

From DEPARTMENT OF MOLECULAR MEDICINE AND
SURGERY

Karolinska Institutet, Stockholm, Sweden

CHARACTERIZATION OF STRUCTURAL CHROMOSOMAL VARIANTS BY MASSIVELY PARALLEL SEQUENCING

Jesper Eisfeldt



**Karolinska
Institutet**

Stockholm 2019

The cover art illustrates a comparison between the chimpanzee (PanTroglodyte5) and human (GRCh38) reference genomes. The human genome is shown at the left-hand side, and the chimpanzee genome is shown mirrored at the right-hand side of the Circos plot (Krzywinski et al. 2009).

The colored ribbons illustrate regions that are shared between the chimpanzee and human genomes; the ribbons are colored by chimpanzee chromosome. Mostly, these large regions (larger than 100,000 base pairs) are structurally conserved between human and chimpanzee; however, there are plenty of differences; for instance, the human chromosome 2 has arisen as a fusion of chimpanzee chromosome 2A and 2B; there is also a large visible inversion of sequence on chromosome 5 – these differences are known as structural variation. A chaotic pattern of thin ribbons can be seen throughout the figure, these ribbons may indicate the movement - or translocation of sequence; but may also indicate regions that are too repetitive for accurate comparison.

The black rectangles forming an inner circle illustrate clusters of protein coding genes that are shared (orthologous) between human and chimpanzee. Notably, the positioning and size of these cluster differ. For instance, there are 5 discrete clusters on human chromosome 18, but only 2 clusters on chimpanzee chromosome 18; in other cases (such as chromosome 13), the clusters appear identical.

All previously published papers were reproduced with permission from the publisher.

Published by Karolinska Institutet.

Printed by Eprint AB 2019

©Jesper Eisfeldt, 2019

ISBN 987-91-7831-589-5

Characterization of structural chromosomal variants by massively parallel sequencing

THESIS FOR DOCTORAL DEGREE (Ph.D.)

By

Jesper Eisfeldt

Principal Supervisor:

Associate professor Anna Lindstrand
Karolinska Institutet
Department of Molecular Medicine and Surgery

Opponent:

Dr Jayne Hehir-kwa
Prinses Maxima Centrum

Co-supervisor (s):

Professor Magnus Nordenskjöld
Karolinska Institutet
Department of Molecular Medicine and Surgery

Examination Board:

Associate professor Maria Wilbe
Uppsala university
Department of Immunology, Genetics and
Pathology

Dr Daniel Nilsson
Karolinska Institutet
Department of Molecular Medicine and Surgery

Associate professor Lena Ström
Karolinska Institutet
Department of Cell and Molecular Biology

Dr Henrik Stranneheim
Karolinska Institutet
Department of Molecular Medicine and Surgery

Associate professor Cecilia Gunnarsson
Linköping University
Department of Cardiovascular Medicine

Dr Valtteri Wirta
Karolinska Institutet
Department of Microbiology, Tumor and Cell
Biology

Dr Francesco Vezzi
Devyser AB

*More than machinery we need humanity,
more than cleverness we need kindness and gentleness.*

Charlie Chaplin, The dictator

ABSTRACT

Chromosomal Structural Variation (SV) such as translocations, inversions, deletions, and duplications are rearrangements of one or several DNA molecules. SVs are widespread across the human genome, and each individual carries thousands of SVs of different types and sizes. SV are known to contribute both to phenotypic diversity and disease traits, and are therefore of interest in multiple fields, including rare diseases research, and clinical diagnostics.

Herein, we present five studies, focused on the analysis of SV using whole genome sequencing (WGS). The project has increased our knowledge regarding the frequency, structure and mechanisms of formation of structural variants in the human genome. In **Paper I, II, and IV**, we develop and evaluate software for detection and analysis of SV using WGS data. In **Paper II, III and IV**, we utilize WGS data to delineate the structure and determine the mechanism of formation of several complex SVs. In **Paper II**, we compare multiple sequencing technologies, and apply these technologies to solve the structure of three complex chromosomal rearrangements. Lastly, in **Paper V**, we validate the use of SV calling from WGS as a routine test in rare disease diagnostics.

Through these studies, we developed and tested tools suitable for WGS SV analysis in a clinical setting. These tools are now part of the routine clinical pipeline; and many of the tools are used by researchers and clinics around the world.

LIST OF SCIENTIFIC PAPERS

- I. **Jesper Eisfeldt***, Francesco Vezzi*, Pall Olason, Daniel Nilsson, Anna Lindstrand. (2017). TIDDIT, an efficient and comprehensive structural variant caller for massive parallel sequencing data. *F1000Research*. 6: p. 664.
- II. **Jesper Eisfeldt***, Maria Pettersson*, Francesco Vezzi, Josephine Wincent, Max Käller, Joel Gruselius, Daniel Nilsson, Elisabeth Syk Lundberg, Claudia M. B. Carvalho, Anna Lindstrand. (2019). Comprehensive structural variation genome map of individuals carrying complex chromosomal rearrangements. *PLOS Genetics*. 15 (2): e1007858.
- III. Lusine Nazaryan-Petersen*, **Jesper Eisfeldt***, Maria Pettersson, Johanna Lundin, Daniel Nilsson, Josephine Wincent, Agne Lieden, Lovisa Lovmar, Jesper Ottosson, Jelena Gacic, Outi Mäkitie, Ann Nordgren, Francesco Vezzi, Valtteri Wirta, Max Käller, Tina Duelund Hjortshøj, Cathrine Jespersgaard, Rayan Houssari, Laura Pignata, Mads Bak, Niels Tommerup, Elisabeth Syk Lundberg, Zeynep Tümer*, Anna Lindstrand*. (2018). Replicative and non-replicative mechanisms in the formation of clustered CNVs are indicated by whole genome characterization. *PLOS Genetics*. 14 (11): e1007780.
- IV. **Jesper Eisfeldt**. Adam Ameer, Daniel Nilsson, Anna Lindstrand. (2019). Discovery of Novel Sequences in 1,000 Swedish Genomes. *Molecular Biology and Evolution*. *Msz176*.
- V. Anna Lindstrand, **Jesper Eisfeldt**, Maria Pettersson, Claudia M. B. Carvalho, Malin Kvarnang, Giedre Grigelioniene, Britt-Marie Anderlid, Olof Bjerin, Peter Gustavsson, Anna Hammarsjö, Patrik Georgii Hemming, Erik Iwarsson, Maria Johansson Soller, Kristina Lagerstedt-Robinson, Agne Lieden, Måns Magnusson, Marcel Martin, Helena Malmgren, Magnus Nordenskjöld, Ameli Norling, Ellika Sahlin, Henrik Stranneheim, Emma Tham, Josephine Wincent, Sofia Ygberg, Anna Wedell, Valtteri Wirta, Ann Nordgren, Johanna Lundin, Daniel Nilsson. From cytogenetics to cytogenomics: WGS as a first line test comprehensively captures the diverse spectrum of disease-causing genetic variation underlying intellectual disability. *Manuscript*

* Equal contribution

LIST OF RELATED SCIENTIFIC PAPERS

- I. Giedre Grigelioniene, Pasi I Nevalainen, Monica Reyes, Susanne Thiele, Olta Tafaj, Angelo Molinaro, Rieko Takatani, Marja Ala-Houhala, Daniel Nilsson, **Jesper Eisfeldt**, Anna Lindstrand, Marie-Laure Kottler, Outi Mäkitie, Harald Jüppner. (2017). A large inversion involving GNAS exon A/B and all exons encoding Gsα is associated with autosomal dominant pseudohypoparathyroidism type 1b (PHP1B). *Journal of Bone and Mineral Research*. 32 (4): 776-783
- II. Maria Pettersson, Raquel Vaz, Anna Hammarsjö, **Jesper Eisfeldt**, Claudia M.B. Carvalho, Wolfgang Hofmeister, Emma Tham, Eva Horemuzova, Ulrika Voss, Gen Nishimura, Bo Klintberg, Ann Nordgren, Daniel Nilsson, Giedre Grigelioniene, Anna Lindstrand. (2018). Alu-Alu mediated intragenic duplications in IFT81 and MATN3 are associated with skeletal dysplasias. *Human mutation*. 39(10):1456-1467
- III. Wolfgang Hofmeister, Maria Pettersson, Deniz Kurtoglu, Miriam Armenio, **Jesper Eisfeldt**, Nikos Papadogiannakis, Peter Gustavsson, Anna Lindstrand. (2018). Targeted copy number screening highlights an intragenic deletion of WDR63 as the likely cause of human occipital encephalocele and abnormal CNS development in zebrafish. *Human mutation*. 39(4):495-505
- IV. Maxime Garcia, Szilveszter Juhos, Malin Larsson, Páll Isólfur Olason, Marcel Martin, **Jesper Eisfeldt**, Sebastian Dilozenzo, Johanna Sandgren, Tomás Diaz, Valtteri Wirta, Monica Nistér, Björn Nystedt, Max Käller (2018). Sarek: A portable workflow for whole-genome sequencing analysis of germline and somatic variants. *bioRxiv*. 316976.
- V. Maria Pettersson*, **Jesper Eisfeldt***, Elisabeth Syk Lundberg, Johanna Lundin, Anna Lindstrand. (2018). Flanking complex copy number variants in the same family formed through unequal crossing-over during meiosis. *Mutation Research/Fundamental and Molecular Mechanisms of Mutagenesis*. 812:1-4
- VI. Adam Ameer, Johan Dahlberg, Pall Olason, Francesco Vezzi, Robert Karlsson, Marcel Martin, Johan Viklund, Andreas Kusalananda Kähäri, Pär Lundin, Huiwen Che, Jessada Thutkawkorapin, **Jesper Eisfeldt**, Samuel Lampa, Mats Dahlberg, Jonas Hagberg, Niclas Jareborg, Ulrika Liljedahl, Inger Jonasson, Åsa Johansson, Lars Feuk, Joakim Lundeberg, Ann-Christine Syvänen, Sverker Lundin, Daniel Nilsson, Björn Nystedt, Patrik KE Magnusson & Ulf Gyllensten. (2017). SweGen: a whole-genome data resource of genetic variability in a cross-section of the Swedish population. *European Journal of Human Genetics*. 5(11):1253-1260
- VII. **Jesper Eisfeldt**, Daniel Nilsson, Johanna Andersson-Assarsson, Anna Lindstrand. (2018). AMYCNE: Confident copy number assessment using whole genome sequencing data. *Plos One*. 26;13(3):e0189710

Equal contribution *

CONTENTS

1	INTRODUCTION	1
1.1	THE HUMAN GENOME	2
1.2	STRUCTURAL VARIATION	4
1.2.1	Mechanisms of SV formation	5
1.2.2	Complex genomic rearrangements	6
1.3	MASSIVELY PARALLEL SEQUENCING	11
1.4	ILLUMINA DYE SEQUENCING.....	12
1.4.1	Paired-end sequencing.....	14
1.4.2	Mate-pair sequencing.....	14
1.4.3	Linked-read sequencing.....	15
1.5	OPTICAL MAPPING	16
1.6	LONG-READ SEQUENCING.....	17
1.7	ANALYSIS OF WGS DATA.....	18
1.7.1	Mapping assembly	19
1.7.2	<i>De novo</i> assembly	20
1.7.3	Quality control and filtering.....	22
1.7.4	Structural variation calling.....	23
1.8	ALGORITHM DEVELOPMENT	29
1.9	PROGRAMMING LANGUAGES	32
2	AIMS OF THE THESIS.....	35
3	MATERIALS AND METHODS.....	37
3.1	COHORT	37
3.2	SHORT-READ SEQUENCING.....	38
3.3	PREPROCESSING OF SHORT-READ WGS DATA.....	39
3.4	ANALYSIS OF BIONANO OPTICAL MAPS	39
3.5	SV ANALYSIS	40
3.6	SNV ANALYSIS.....	40
3.7	SOFTWARE AND PIPELINES	41
3.7.1	TIDDIT and SVDB.....	41
3.7.2	FindSV	42
3.8	STATISTICAL ANALYSES	43
3.9	MOLECULAR ANALYSES.....	44
3.9.1	Array comparative hybridization (aCGH).....	44
3.9.2	Fluorescence in situ hybridization (FISH)	44
3.9.3	Karyotyping.....	44
4	RESULTS	45
4.1	FindSV	45
4.2	COMPLEX GENOMIC REARRANGEMENTS.....	45
4.3	COMPARISON OF WGS TECHNOLOGIES	45
4.4	NOVEL SEQUENCES IN THE SWEDISH POPULATION.....	46
4.5	AS A FIRST-TIER CLINICAL TEST IN GENETIC DIAGNOSTICS	47
5	FUTURE PERSPECTIVES.....	48
6	ACKNOWLEDGEMENTS.....	49
7	REFERENCES.....	53

LIST OF ABBREVIATIONS

ACMG	American College of Medical Genetics
ANOVA	Analysis of variance
aCGH	Array comparative hybridization
BCL	Base call file
BFB	Breakage-fusion-bridge
bp	Base pair
BWA	Burrows-Wheeler aligner
CCD	Charge-coupled device
CGR	Complex genomic rearrangement
CNV	Copy number variation
DBSCAN	Density-based spatial clustering of applications with noise
DNA	Deoxyribonucleic acid
FISH	Fluorescent <i>in-situ</i> hybridization
FoSTeS	Fork Stalling and template switching
FM	Full-text index in minute space
HGVS	Human genome variation society
Kbp	Kilo base pair
LINEs	Long interspersed nuclear elements
MP	Mate-pair
MPS	Massively parallel sequencing
NAHR	Non-allelic homologous recombination
NHEJ	Non-homologous recombination
PE	Paired-end
QC	Quality control
RAM	Random access memory
RNA	Ribonucleic acid
SNV	Single nucleotide variant
SV	Structural variation
UPD	Uniparental disomy
WES	Whole exome sequencing
WG	Whole genome
WGS	Whole genome sequencing

1 INTRODUCTION

Bioinformatics is an interdisciplinary field, applying computer science and statistics to answer biological questions. Bioinformatics is a relatively young field: the term was coined in the 1970s by Paulien Hogeweg and Ben Hesper, and was not popularized until the emergence of high throughput biology in the 1990s (Hogeweg 2011). The field has grown steadily since the emergence of high throughput biology, mainly due to the rapid generation of large amounts of data at a low cost; this phenomenon is exemplified with the early human genome project, where so called shotgun sequencing was applied to read and analyze millions of short DNA fragments (Venter et al. 1998).

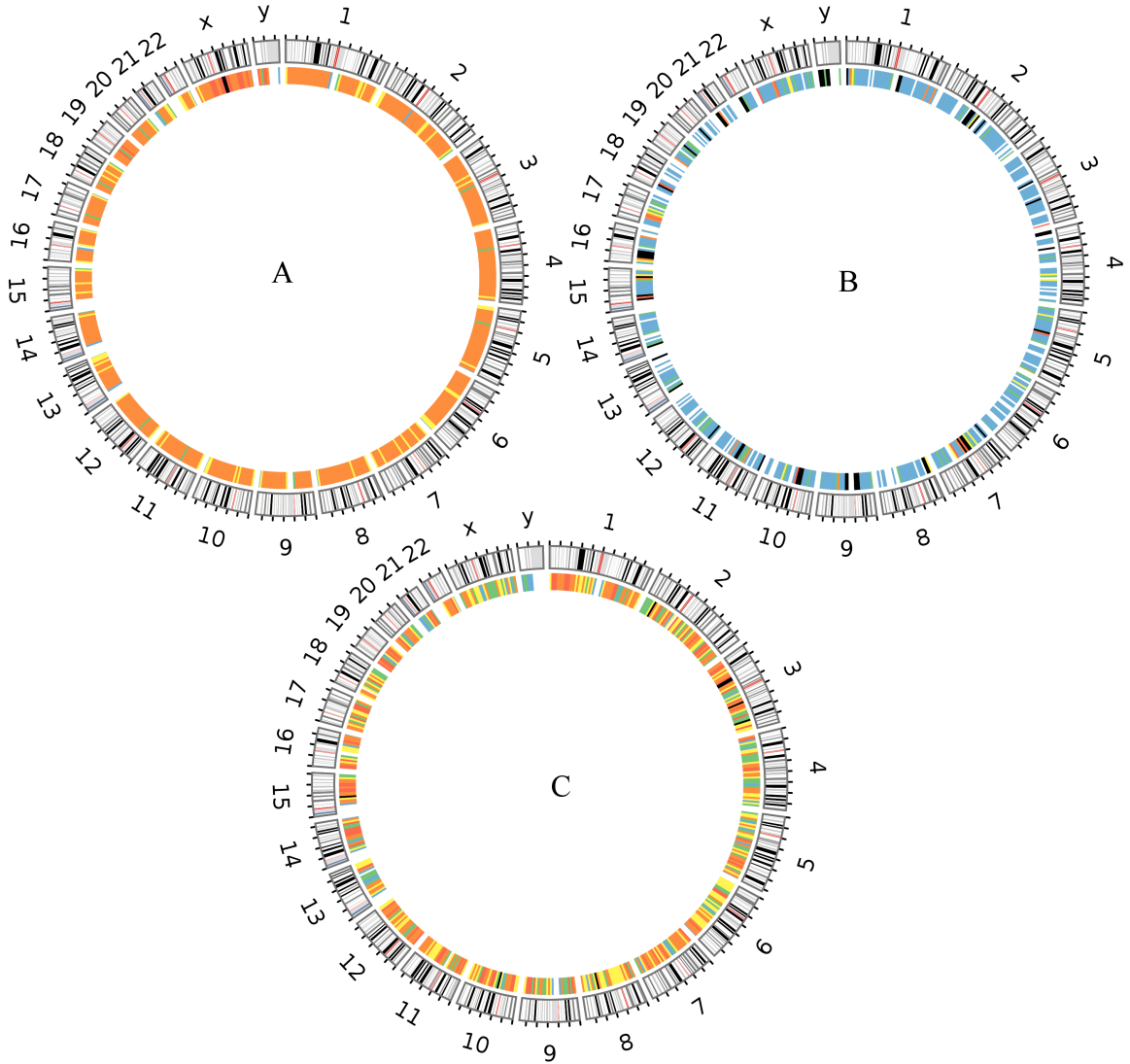
Today, Bioinformatics is present in a wide range of biological disciplines, including Environmental biology (Zeigler Allen et al. 2017), Genomics (Mikkelsen et al. 2005), and Systems biology (English et al. 2011). Bioinformaticians work with a great variety of tasks, including data mining (Bolser et al. 2017), image analysis (van der Donk et al. 2018), and software development (Li 2013). In everyday communication, a bioinformatician is a researcher or clinician working primarily with any of the topics covered by bioinformatics. As such, bioinformaticians come from variety of backgrounds, and includes biostatisticians, software developers, and molecular biologists.

This thesis discusses the bioinformatics necessary to analyze structural variation (SV) using whole genome sequencing (WGS) data in a clinical setting; including the development and benchmarking of software, as well as evaluation of massively parallel sequencing (MPS) technologies.

1.1 THE HUMAN GENOME

The first human genome was sequenced through the Human genome project (Lander et al. 2001) and by the Celera corporation (Venter et al. 1998); providing a draft reference genome. This draft genome consisted of 24 haploid chromosomes 1-22, X and Y, and totaled 3.08 billion bases (Collins et al. 2004). Notably, the human genome is highly repetitive: 66% of the human genome consists of so-called repeat elements (de Koning et al. 2011) – short sequences of DNA present in multiple copies across the genome (Figure 1A). Additionally, 5% of the human genome consists of segmental duplications – sequences longer than 1 Kilo base pairs (Kbp) that are present in more than two copies in the haploid genome (Sharp et al. 2005) (Figure 1B). The number of human genes is not fully known, and estimates vary between 26,000 – 68,000 genes – mainly due to different methods for detecting and classifying the genes, as well as different definitions on what genes are (Salzberg 2018); roughly 21,000 of these genes encode for proteins (Salzberg 2018), the remainder encode for various types of regulatory RNA (Morris and Mattick 2014), or pseudo genes – genes that are duplicated and then mutated beyond their original function (Zheng et al. 2007). Although the genes cover roughly 50% of the genome (Figure 1C), only 2% of the genome consist of coding regions (Kinsella et al. 2011) – the remainder is believed to have regulatory (Dunham et al. 2012), structural (Sahlén et al. 2015), no significant, or unknown function.

Figure 1. Circos heatmap plots illustrating the density of A) repeat elements, B) segmental duplications, and C) protein coding genes. The densities are computed for bins sized 3 Mbp, spread evenly across the human genome. The color indicates the percentage coverage of each feature within each bin; white < 0.5%, blue < 15%, green < 30%, yellow < 45%, orange < 60%, red < 75%, and black > 75%.



The total cost of the human genome project was 3 billion dollars; since then, the sequencing cost has decreased steadily, and today, a human genome may be sequenced for less than 1000\$, allowing for routine clinical applications (Hayden 2014). This decrease in sequencing cost is explained by the rapid development of the MPS platforms, offering higher throughput at a lower cost. The human reference genome has improved greatly since the completion of the human genome project (Schneider et al. 2017); however, the majority

(70%) of the sequence originates from only one individual; as such, there's a lack of population specific sequence (Seo et al. 2016). A number of nationwide population genomic studies have been now performed and these studies indicate that there is a great genetic variability among individuals and populations, and that there is a great benefit in creating so called local reference genomes (Boomsma et al. 2014; Ameer et al. 2017; Maretty et al. 2017).

1.2 STRUCTURAL VARIATION

Structural variation (SV) is genetic variation covering at least 1 Kbp. There's an abundance of SV in the human genome where on average every individual carry roughly 10,000 SVs (Sudmant et al. 2015). Taken together, SVs comprise the largest proportion of sequence variation between individuals (Sudmant et al. 2015), and act as an important contributor to the evolution of the human genome (Prüfer et al. 2012; Lupski 2015) Therefore, it is not surprising that SVs are important contributors to the phenotypic traits of human individuals (Lupski 2015). There is a great diversity of SV: SV may be unbalanced, *i.e.* involving the gain or loss of sequence, or balanced involving no gain or loss of sequence (Lupski 2015). Unbalanced SV comprise of deletions (loss of one or more copies), duplication (gain of a single copy), as well as amplification (consisting of gains of more than one copy of sequence) (Redon et al. 2006). Duplications are commonly classified based on the positioning and orientation of the duplicate copy. Additionally, sequence may be inserted into the human genome through viral infection (Vogt 1997) or mutation (Sherman et al. 2019), such SV are known as novel sequence insertions.

Balanced SVs are divided into two categories: translocations (*i.e.* repositioning of sequence) (Scriven et al. 1998) and inversions (Feuk 2010). Translocations may be further divided into interchromosomal translocations (exchange between chromosomes) or intrachromosomal translocations (repositioning of sequence within a single chromosome) (Tümer et al. 1992).

SVs are commonly described using the human genome variation society (HGVS) nomenclature (Taschner and den Dunnen 2011) and its extension, implemented in the ISCN nomenclature (ISCN 2016), allowing for standardized description of all SVs in the human genome.

1.2.1 Mechanisms of SV formation

SVs are formed as a consequence of double stranded DNA breaks (McClintock 1941) or through errors during DNA replication (Lupski 2015). These breaks and errors are repaired using a variety of pathways; different enzymes are present at different stages of the cell cycle, and various enzymes act on different features of the genome. Therefore, these mechanisms produce SV carrying different characteristic (Haber 2000); these characteristics can be recognized by analyzing the breakpoint junctions (Weckselblatt and Rudd 2015).

Fork stalling and template switching (FoSTeS) is an example of a replicative SV formation mechanism (Gu et al. 2008). FoSTeS may occur if the DNA polymerase is stalled during active DNA replication; the DNA polymerase is more likely to be stalled in regions where the DNA is likely to form secondary structures such as hairpins (Zhang et al. 2009). Once DNA replication is stalled, nearby replication forks may switch DNA templates using complementary template microhomology, resulting in the replication of aberrant chromosomes (Lee et al. 2007). FoSTeS is commonly characterized by small templated insertions at the breakpoint junction of SV (Weckselblatt and Rudd 2015).

Non-allelic homologous recombination (NAHR) is another SV formation mechanism. NAHR is performed using the same enzymatic machinery as homologous recombination; however, instead of involving alleles, NAHR involves non-allelic regions sharing high sequence similarity (Shaw 2004). NAHR may occur due to misalignment of the chromatids during meiosis (Gu et al. 2008), resulting in recombination between homologous and non-allelic sequences. However NAHR may also occur as a mechanism of repairing double

stranded DNA breaks (Mao et al. 2008). NAHR may occur throughout the entire cell cycle, but is mostly occurring throughout the S and G2 phases (Mao et al. 2008). Similar to homologous recombination, NAHR can only occur between homologous regions. Such regions include segmental duplications and repeat elements such as long interspersed nuclear elements (LINEs). SVs located within such regions are therefore likely to be formed through NAHR (Weckselblatt and Rudd 2015).

Non-homologous end joining (NHEJ) is a mechanism utilized to repair double stranded DNA breaks (Haber 2000); NHEJ is one of the major repair mechanisms in eukaryotic organisms, and may occur throughout the entire cell cycle (Mao et al. 2008). NHEJ utilize short (5-25 bp) stretches of microhomology to guide the fusion of double stranded DNA breaks. SVs formed through NHEJ can therefore be recognized by the presence of short stretches of microhomology at the breakpoint junction (Weckselblatt and Rudd 2015).

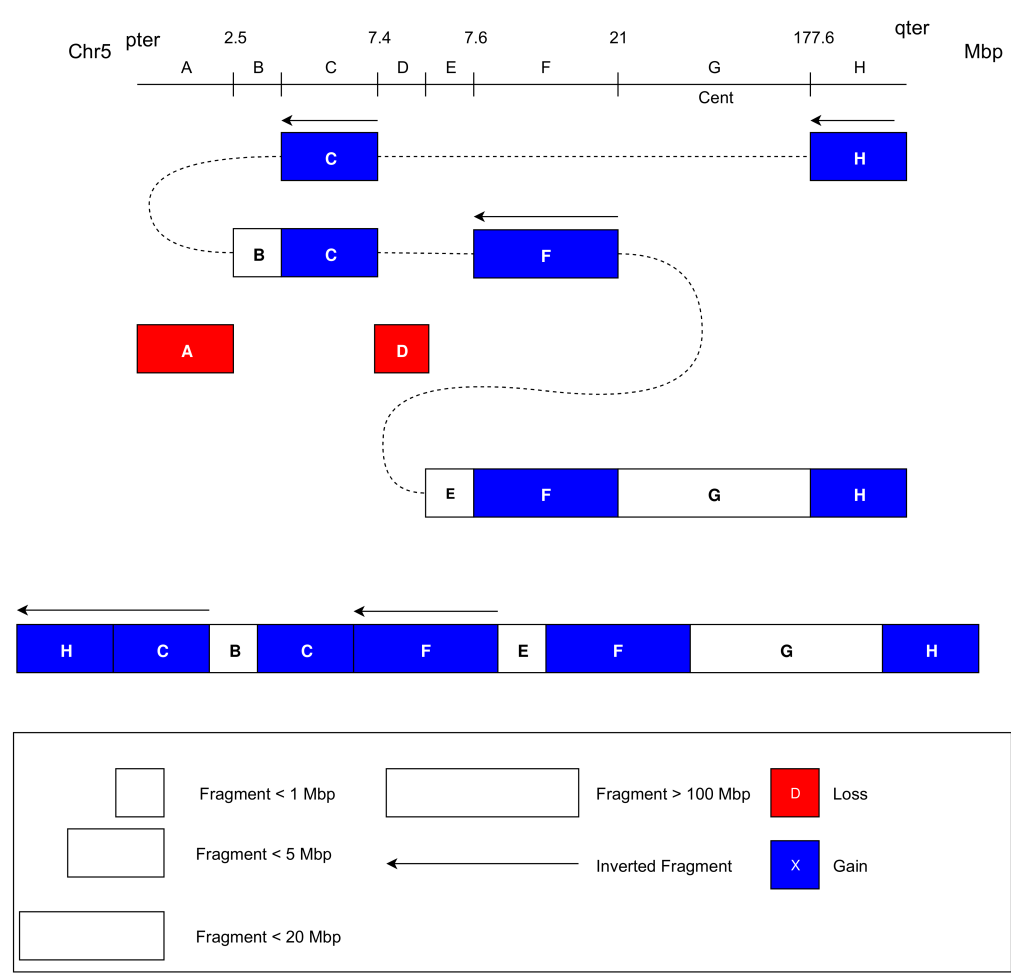
1.2.2 Complex genomic rearrangements

Complex genomic rearrangements (CGRs) are genomic rearrangements consisting of at least two adjacent breakpoint junctions. CGRs are commonly found in cancer genomes. Germline pathogenic CGRs are rare phenomena causing a variety of disorders, including intellectual disabilities and malformation (Collins et al. 2017). However, CGRs may be phenotypically neutral, and small CGRs are found in every individual (Collins et al. 2019). CGRs arise through a diversity of mechanisms; these mechanisms give rise to characteristic rearrangements that can be identified by analyzing a variety of features, including copy number changes, orientation of the fragments, and the clustering of the breakpoints (Pellestor 2019).

Discovered in the late 1930s, *the Breakage-Fusion-Bridge cycles* (BFB cycles) is the first mechanism describing the formation of CGR (McClintock 1938; McClintock 1941). BFB cycles may be initiated through telomeric attrition. Chromatids lacking a telomere may fuse with other chromatids; during anaphase, these fused chromatids will be pulled apart:

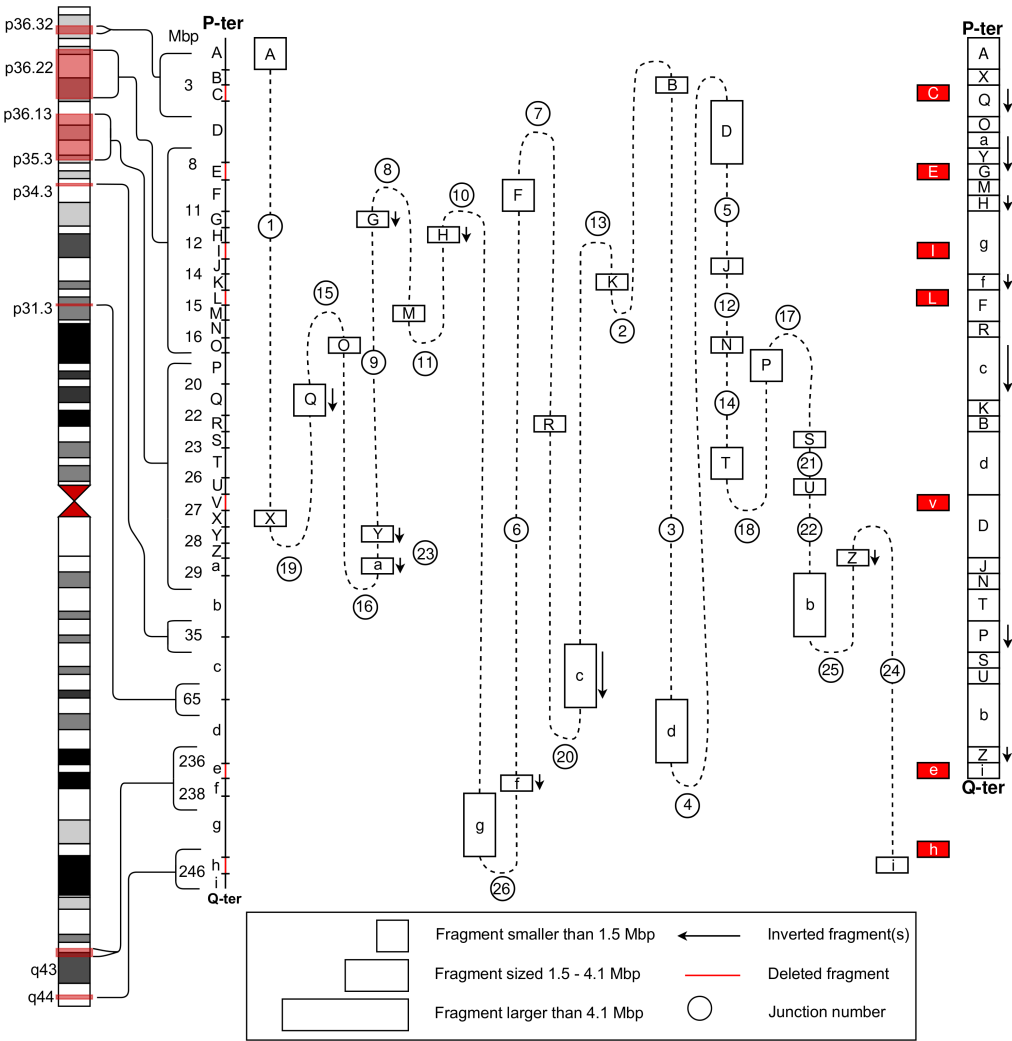
resulting in two rearranged chromatids lacking telomere, allowing for multiple cycles of bridging, fusion and breakage (McClintock 1941). The BFB cycle will end only if a telomere is gained, or if the aberrant chromosome forms a ring chromosome. CGRs produced through BFB cycles are therefore characterized as terminal rearrangements, often including a terminal deletion (Figure 2). CGRs formed through BFB cycles will have an overrepresentation of fragments fused in head-to-head, tail-to-tail configurations (Kinsella and Bafna 2012). Each fusion is an independent event, the breakpoints are therefore likely to not cluster, and the resulting CGR can be recognized as the product of a multistep process. BFB-cycles may give rise to any copy number states, including deletions and duplications, or entirely copy number neutral CGRs (Zakov and Bafna 2014).

Figure 2. A CGR formed through BFB-cycles. The rearrangement consists of the deletion of the chromosome 5 p-terminal and duplication of the q-terminal, followed by characteristic inverted duplications. The rearrangement is presented in **Paper III**.



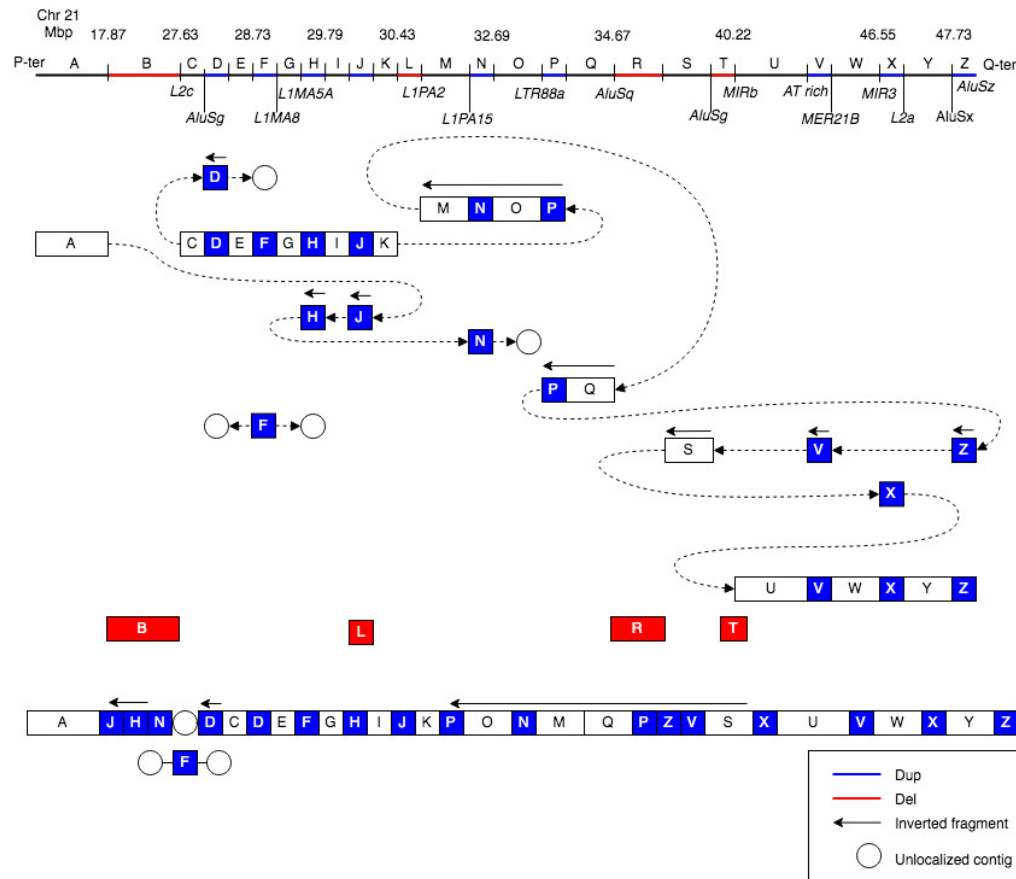
Chromothripsis is another well-known mechanism: chromothripsis is a cataclysmic event caused by instantaneous and localized pulverization of one or a few (<4) chromosomes or chromosome arms. This pulverization may be caused by various events, including viral infection, errors during cell division, or radiation (Koltsova et al. 2019); recently, chromothripsis has been shown to occur through the formation of micronuclei (Zhang et al. 2015). Once shattered, the resulting DNA fragments are either degraded or fused into aberrant chromosomes by the DNA repair machinery (Pellestor 2019). The CGRs formed by chromothripsis will therefore involve a large number (>10) of clustered breakpoints; the fragments are fused and degraded randomly resulting in randomly positioned and oriented fragments (Zhang et al. 2009) including both deletions and copy number neutral segments (Korbel and Campbell 2013) (Figure 3).

Figure 3. A CGR formed through chromothripsis. A chromotriptic rearrangement of chromosome 1, with breakpoints clustering on p36.22-p35.3; the rearrangement consists of 34 fragments. The figure is adapted from **Paper II**.



Chromoanasythesis occurs due to replicative stress (Liu et al. 2011) and such stress may be due to endogenous factors, including DNA secondary structures, as well as exogenous factors, including radiation. Once stalled, the replication fork may undergo a series of error prone repair mechanisms, including FoSTeS, resulting in aberrant chromosomes (Pellestor 2019). Chromoanasythesis is known to involve a large number of chromosomes and chromosome arms (>3), the breakpoints are not necessarily clustered, and reflect the 3D structure of the DNA at the time point of the event. Usually, chromoanasythesis involve a small number (<20) of dispersed breakpoints (Figure 4). Chromoanasythesis may give rise to any copy number events, including amplifications – and the type of event reflect the path taken by the DNA polymerases replicating the chromosomes (Korbel and Campbell 2013; Zepeda-Mendoza and Morton 2019).

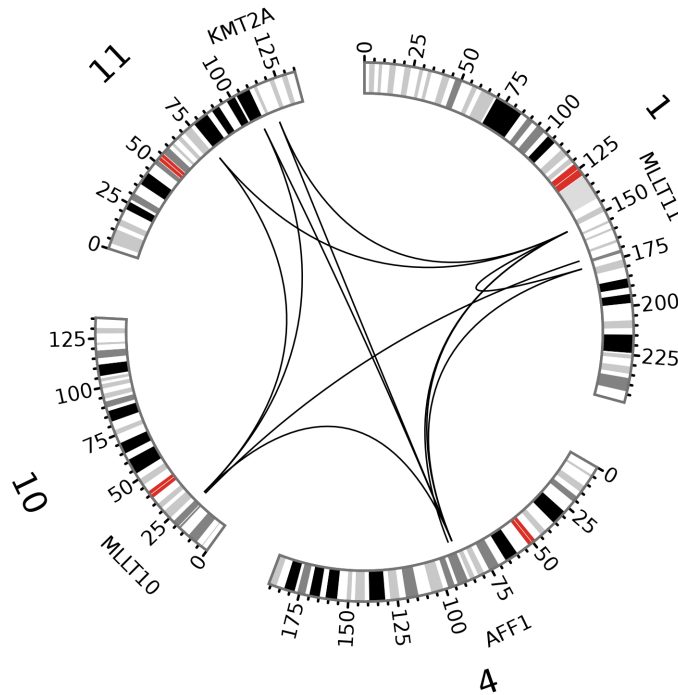
Figure 4. A CGR formed through chromoanasythesis. A complex rearrangement on chromosome 21; the rearrangement includes duplications (blue), as well as deletions (red). The derivative chromosome is shown below. The figure is adapted from **Paper III**.



Lastly, *chromoplexy* is a newly reported mechanism of CGR formation (Baca et al. 2013). Chromoplexy is caused by catastrophic errors during transcription (Haffner et al. 2010), resulting in aberrant chromosomes similar to, but distinct from those formed by chromothripsis. Co-regulated genes are brought close together during active transcription, forming so called transcription hubs (Babu et al. 2004). During transcription, the chromatin is opened and the DNA is made accessible and unprotected (Ehrenhofer-Murray 2004); the DNA is therefore more likely to break. Chromoplexy is caused by breakage of the DNA in such transcription hub, followed by incorrect repair by the broken DNA strand (Haffner et al. 2010). Chromoplexy may occur as a single catastrophic event similar to chromothripsis, but may also occur as a chain process (Zhang et al. 2013).

CGRs formed by chromoplexy may therefore be characterized based on the positioning of the breakpoints, the breakpoints should be located within and around genes that are co-regulated. The breakpoints could be clustered, but also non-clustered, and the positioning of the breakpoints reflects on the 3D structure of the transcription hub; commonly chromoplexy include a large (>2) number of chromosomes. Compared to chromothripsis, the breakpoints are usually fewer and more dispersed. Copy number variation (CNV) is atypical to chromoplexy, and the CGRs are entirely balanced except for small deletions at the breakpoint junctions (Zepeda-Mendoza and Morton 2019). Figure 5 illustrates an example of a CGR formed through chromoplexy; the CGR is balanced, and consists of 24 breakpoints spread across four chromosomes (1, 4, 10, 11). The majority of the breakpoints are located close to genes.

Figure 5. A CGR formed through chromoplexy illustrated as a Circos plot. The gene names (*AFF1*, *KMT2A*, *MLLT10*, *MLLT11*) indicate genes affected by the rearrangement. The black arcs illustrate the breakpoint junctions of the rearrangement.



1.3 MASSIVELY PARALLEL SEQUENCING

MPS is a term used to describe any technology used to sequence (*i.e.* to read) a large number (commonly millions) of DNA fragments (Shendure and Ji 2008) in parallel. MPS may be used to sequence the entire genome of an organism (Venter et al. 1998), such experiment is known as WGS. Additionally, MPS may be used to sequence selected regions of interest (Albert et al. 2007), such as the whole exome (WES), or a group of known disease genes (usually known as a gene panel). Recently, MPS has also been applied to sequence RNA (Brenner et al. 2000), analyze epigenetic markers (Johnson et al. 2007), DNA conformation (Dekker et al. 2002), and to quantify protein levels (Stoeckius et al. 2017).

The first commercially available MPS platform (Genome Sequencer 20) was presented in 2005 by Roche (Margulies et al. 2005), and a great diversity of sequencing platforms has been developed since (Barba et al. 2013). Each MPS platform utilize different biochemical

properties of the DNA (Healy 2007) or DNA replication (Purushothaman et al. 2006) to obtain the DNA sequence. Notably, each of these platforms perform differently, providing DNA sequences of different length, quality, and cost (Barba et al. 2013); as such, there is a great need of comparing various platforms in order to find the optimum method for a given task. Today, MPS is used for the diagnostics of a variety of diseases, including inborn diseases such as intellectual disabilities and malformations (Bowling et al. 2017), as well as acquired diseases, such as cancer (Laduca et al. 2014). MPS platforms are usually divided into two categories: short-read sequencers and long-read sequencers. The short-read sequencers have a higher throughput, and are more accurate compared to the long-read sequencers (McNaughton et al. 2019). On the other hand, the long-read sequencers produce longer reads (Lee et al. 2016), which is necessary for analyzing the repetitive regions of the genome. As of date, there are three major commercially available providers of short-read sequencing technologies: Illumina dye sequencing (Bennett 2004) (first marketed by Solexa and acquired by Illumina Inc), semiconductor sequencing (Purushothaman et al. 2006) (marketed by Thermo Fisher under the brand Ion Torrent), and DNA nanoball sequencing (Drmanac et al. 2010) (commercialized by Complete Genomics and acquired by Beijing Genomics Institute (BGI)); as well as numerous smaller providers, such as Qiagen (<https://www.qiagen.com/us/>).

1.4 ILLUMINA DYE SEQUENCING

Illumina dye sequencing dominates the sequencing market due to the low cost, low error rate and high throughput of their platforms (Quail et al. 2012). The platform is applied to a wide range of applications, including RNA sequencing (Nagalakshmi et al. 2008), DNA sequencing (Sudmant et al. 2015), and various epigenetic analyses (Barski et al. 2007). Typically, the read length is set to 151 bp, but the read length may vary from 25 to 300 bp depending on application and platform (Quail et al. 2012). The quality of the sequenced

bases decreases towards the end of the read (Dohm et al. 2008), therefore, it is typically not practical to sequence reads longer than 151 bp. Today, the standard human WGS sample consists of roughly 300 million pairs of reads, resulting in 30X average coverage across the genome. However, higher coverage may be desired in the analysis of cancer samples (Griffith et al. 2015). For economic reasons, a lower coverage may be necessary when analyzing large populations (Gusev et al. 2012; Sudmant et al. 2015; Chiang et al. 2018; Liu et al. 2018).

Illumina dye sequencing is performed in a stepwise manner, the DNA is first prepared to a library, next clonally amplified to increase signal strength, and finally sequenced. There are a wide range of library preparation protocols. Briefly the DNA is purified, fragmented, and a sequencing adapter is ligated to each end of the DNA fragments (Rhodes et al. 2014). The sequencing adapter is a small DNA fragment designed to hybridize with complementary DNA sequences on the surface of a small glass chip, the flow cell (Holt and Jones 2008). Once the DNA is hybridized to the flow cell, each DNA fragment is amplified into thousands of identical copies. This process is known as clustering. Once clustering is complete, complementary sequences are separated and rinsed from the flow cell, such that only single stranded DNA fragments of the same strand remains (Balasubramanian 2011). Now the flow cell is ready for sequencing; the flow cell is filled with a solution containing DNA polymerase and nucleotides. Historically, these nucleotides carry a fluorescent label specific to each of the four nucleotides (A, C, T, G); however, more recent platforms use a two channel system where C and T are represented as red and green; A is represented as a mix of green and red, and G is unlabeled (Illumina Inc 2013). Additionally, each nucleotide carries a reversible 3' blocker, preventing the addition of multiple nucleotides (Balasubramanian 2011).

The DNA polymerases will add the complementary nucleotide to each fragment in every cluster on the flow cell. The solution is thereafter washed away, leaving the DNA

fragments as well as the recently added fluorescent label. The fluorescence of each cluster is captured using a charge-coupled device (CCD) camera, thereby recording which nucleotide was inserted. The fluorescent label, as well as the 3' blocker is then cleaved from the fragments, and washed away from the flow cell. This procedure is repeated for a number of cycles, resulting in reads of the same length as the number of cycles (Balasubramanian 2011). The resulting measurements are stored in a base call file (BCL).

1.4.1 Paired-end sequencing

The Illumina sequencing platforms may be used to generate reads of 25-300bp in length. It is impractical to generate longer reads, since the base qualities decrease for each sequenced base of the read (Dohm et al. 2008). Instead, it is common to sequence both ends of a longer DNA fragment (300-550bp), so called paired-end (PE) sequencing (Fullwood et al. 2009). The process of sequencing both ends is initially the same as when sequencing a single read. However, once the first read is sequenced, another round of bridge-PCR is initialized. Through this reaction, the complementary strand of the initially sequenced strand is regenerated; allowing for the sequencing of both ends of the fragments (Illumina Inc 2009). PE sequencing data is useful in many settings, including SV detection and *de novo* assembly (Korbel et al. 2007). In particular, the PE reads may be used to align the reads uniquely close to repeat regions; resulting in a higher horizontal coverage across the genome (Lander et al. 2001). However, single read sequencing is cheaper compared to PE sequencing, and may therefore be preferred for some applications (Griffith et al. 2015).

1.4.2 Mate-pair sequencing

PE sequencing is used to sequence both ends of DNA fragments sized (300-550 bp), which increases the horizontal coverage across the genome (Venter et al. 1998). However, 300-550 bp is short compared to many of the repeats within the human genome (Sharp et al.

2005), and larger insert sizes are therefore required to overcome such repeats. However, fragments larger than 1 Kbp cluster poorly, resulting in low yield and low quality (Bronner et al. 2014; Tan et al. 2019).

This limitation may be overcome through mate-pair (MP) sequencing (van Heesch et al. 2013). In MP sequencing protocols, the DNA is sheared into larger fragments (3-100 Kbp), the resulting fragments are size selected, and circularized. Non-circularized DNA is removed by digestion; resulting in a circularized DNA library consisting of fragments of similar sizes. The circularized DNA is cleaved open into fragments of 300-600 bp, such that the ends of the original fragments now are the ends of the resulting smaller fragment (Illumina 2016). The resulting fragments are therefore of small enough size to produce clusters of high quality, yet they provide long range information not present in the standard PE experiment (Van Nieuwerburgh et al. 2012).

Although MP sequencing allows for the resolving of repetitive regions (Kelley and Salzberg 2010), the library-preparation is lengthy and expensive; additionally, the resulting libraries may suffer from low complexity (few unique molecules), and may therefore yield a higher duplication rate when sequenced (Van Nieuwerburgh et al. 2012). Therefore, it is preferred to perform shallow sequencing using MP protocols, combined with deeper sequencing using standard short PE sequencing (van Heesch et al. 2013; Maretty et al. 2017).

1.4.3 Linked-read sequencing

Long-range information is useful in both research and clinical settings, and is necessary for the phasing of genetic variation in cis, as well as detection of SV located in repetitive regions (Cretu Stancu et al. 2017). Linked-read sequencing provides long-range information using the Illumina short-read sequencing platforms, and is therefore cost effective compared to current single molecule sequencers (Zheng et al. 2016). As of date, 10X genomics is the only commercially available solution of linked-read sequencing.

Linked-read sequencing is however a widely researched topic, and there are many solutions being developed (Redin et al. 2017; Zhang et al. 2017). All these methods have in common that the long DNA molecules are separated in droplets, with ideally one molecule per droplet. Inside this droplet, the molecules are fragmented and a barcode is added to each fragment. This barcode is unique for each droplet, and is therefore shared with all fragments originating from that droplet. The fragments, along with their barcodes are then sequenced on an Illumina sequencer, using a standard PE protocol (Zheng et al. 2016; Redin et al. 2017; Zhang et al. 2017).

The reads are later analyzed using specialized softwares that group (or link) the reads according to barcode to form longer chains of reads *in silico* (Weisenfeld et al. 2017).

These synthetic long molecules may be used to phase SNVs, call SVs (<https://github.com/10XGenomics/longranger>), and improve *de novo* assemblies (Jackman et al. 2018).

1.5 OPTICAL MAPPING

Optical mapping is a technology used to create genomic maps through fluorescent labeling of long DNA molecules (Schwartz et al. 1993). Today, Bionano genomics (Cao et al. 2014) is the only commercially available solution; however, there is a great variety of approaches in literature, including the OPGen Argus platform (Onmus-Leone et al. 2013), which was discontinued in 2015 (www.opgen.com). An optical map is created by introducing fluorescent probes into long DNA molecules at specified DNA motifs (*i.e.* short sequences of DNA). These probes are subsequently imaged using fluorescent microscopy (Schwartz et al. 1993; Onmus-Leone et al. 2013; Cao et al. 2014). The resulting maps are error corrected and assembled *in-silico*, resulting in long consensus maps consisting of multiple molecules; such consensus maps may span entire chromosome arms (Deschamps et al. 2018), and usually result in the assembly of discrete haploblocks (Chan et al. 2018). Lastly, the

resulting maps are used for hybrid-assembly (Deschamps et al. 2018) or SV detection (Cao et al. 2014).

1.6 LONG-READ SEQUENCING

Long-read sequencing is a term used to describe technologies used to produce long-reads (average >10 Kbp) (Lee et al. 2016). Long-reads are desirable in order to span repetitive regions and to characterize the structure of the chromosomes; this is important for SV detection (Cretu Stancu et al. 2017), phasing of the genomes (Chan et al. 2018), as well as for the creation of reference genomes (Deschamps et al. 2018). As of date, there are two commercially available long-read sequencing technologies: Oxford Nanopore (Stoddart et al. 2009), as well as Pacific Biosciences SMRT (Levene et al. 2003); however, there is a diversity of experimental long-read technologies, including electron-microscopy sequencers (Michalet et al. 1997; Bell et al. 2012) and graphene nanopore sequencers (Haque et al. 2013). Long-read sequencing allows for the direct sequencing of single molecules, which is advantageous as it allows for the characterization of full transcripts (Depledge et al. 2019), reduces various sequencing biases (Goldstein et al. 2019), and allows for the analysis of epigenetic markers (Gigante et al. 2019).

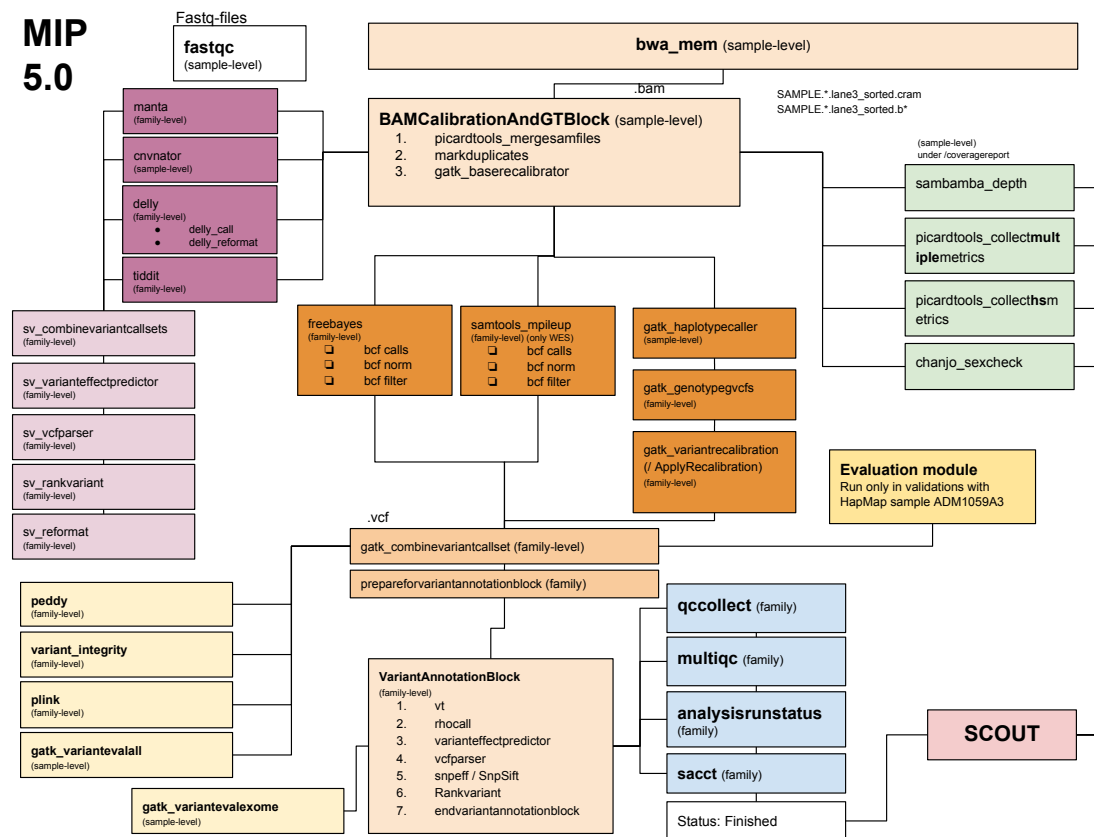
Despite these advantages, the use of long-read sequencing is limited. The error rate of long-read sequencing is high, and commonly, there is one sequencing error for every tenth base (Rang et al. 2018). Long-read sequencing is typically at least three times as expensive per base pair, and high depth may be required in order to compensate for the high error rate (Rang et al. 2018). The preparation of long-read sequencing libraries are challenging compared to short-read sequencing libraries: the DNA fragments are fragile, and too fragmented DNA will provide poor data, conversely, long DNA molecules are prone to worsen the throughput of the sequencers (Schalamun et al. 2019).

1.7 ANALYSIS OF WGS DATA

The analysis of sequencing data is performed in a stepwise manner. These steps are usually performed by pipelines. A pipeline is a software that controls and runs other software in an organized and reproducible manner and many such pipelines are available. Some are focused on smaller tasks, such as preprocessing of sequencing data or variant calling (<https://github.com/J35P312/FindSV>), while others, perform the entire end-to-end WGS analysis (Figure 6) (Stranneheim et al. 2014). The majority of the pipelines are focused on a single sequencing technology; and most of the pipelines are custom made to suit the needs of the group or company that developed it.

Regardless of the pipeline or sequencing platform, assembly (either *de novo* or by mapping to a reference) is one of the first steps of the analysis (Langmead et al. 2009). The assembly process is then followed by quality control (QC), and filtering, or labeling of low-quality data. Once the preprocessing is complete, variant analysis may be initiated. The following sections describes the analysis of WGS data.

Figure 6. A flowchart representing the WGS analysis performed by the mutation interpretation pipeline (MIP). The MIP pipeline, used by Clinical Genomics Stockholm, accepts fastq files as input, and returns a variety of outputs, including variant calls and QC metrics.



1.7.1 Mapping assembly

Mapping assembly (or simply alignment) is commonly used when working with human whole genome data. In part because of its ease of use, but also due to lower computational cost compared to *de novo* assembly, which is the only alternative method (Ekblom and Wolf 2014).

Alignment is the process of mapping the sequenced reads to a reference genome (Langmead et al. 2009). This process is performed using software called aligners or mappers. Today BWA (Li 2013) is the most commonly used aligner. However, there are a large number of aligners available. These include Novoalign (<http://www.novocraft.com>), SOAP2 (R. Li et al. 2009), and Bowtie (Langmead et al. 2009), as well as a large number of

less commonly used tools. The aligners use different algorithms and statistics to align the sequencing data to the reference genome.

Additionally, the aligners add non-standard information using different formats to their output binary alignment map (BAM) (H. Li et al. 2009) file; this information may be necessary in downstream analysis such as variant calling. The licensing of the tools is another factor to consider, for instance, BWA is open source, while Novoalign is commercially developed. Hence the choice of aligner is not only depending on the performance of the aligner, but also on the downstream analysis, and the setting which the analysis takes place (*i.e.* academic or commercial). Due to the importance of sequence alignment, a number of studies have been conducted to compare the performance of the aligners. In general, different aligners perform differently on different genomes and sequencing technologies. On Illumina data, Novoalign has the greatest overall sensitivity while aligners utilizing the full-text index in minute space (FM) index has the smallest computational demand (Shang et al. 2014; Thankaswamy-Kosalai et al. 2017).

1.7.2 *De novo* assembly

De novo assembly is the second fundamental method for assembling the WGS reads. *De novo* assembly is performed by merging similar reads into longer contiguous sequences (Myers et al. 2000). These contiguous sequences are commonly known as contigs, and serves as the basis for performing a wide range of tasks, including variant calling (Li 2015), transcriptome assembly (Grabherr et al. 2011), and creation of reference genomes (Myers et al. 2000; Lander et al. 2001). Most *de novo* assemblers follow two distinct approaches: the overlap approach (Hernandez et al. 2008), or the De Bruin graph approach (Zerbino and Birney 2008). The overlap approach is rather intuitive: The assembler compares all reads against each other, and merges reads that satisfy a certain overlap threshold. This process may be performed in various ways, commonly, an overlap graph or table is constructed. These data structures describe the amount of overlap between the reads; once constructed,

these data structures are simplified and converted into contigs (Peltola et al. 1984). Due to the large number of reads in a typical WGS experiment (600 millions), an all versus all read comparison is no longer feasible, as such, the most moderns overlap assemblers utilize indices, allowing the reads to be compared at a smaller time complexity (Li 2012; Simpson and Durbin 2012)

The De Bruin graph method is fundamentally different from the overlap method; instead of comparing the reads directly, the reads are separated into substrings of a length commonly specified by the user (Simpson et al. 2009). These substrings are known as k-mers, and all substrings of a specified length will be extracted from each read. As such there will be 89 k-mers per read, if the read length is 150, and the k-mer length is set to 61. The assembler will analyze every read, and extract every k-mer, and simultaneously, it will create a De Bruin graph. In this graph, each k-mer specifies a node, and each vertex specify the overlap between k-mers. Once the graph is constructed, the graph is simplified and converted into contigs (Zerbino and Birney 2008). Through this process, no all-versus all comparison is performed, as such, the De Bruin graph approach is potentially faster than the overlap approach, on the other hand, the resulting assembly may be less contiguous due to the short k-mer length (Chopra et al. 2014), and some De Bruin assemblers are prone to utilize large amounts of memory due to the size and complexity of the graph (Khan et al. 2018).

Upon finishing the assembly, one can continue with downstream analyses, such as QC (Gurevich et al. 2013), or variant calling (Li 2012). However, it is also common to increase the contiguousness of the assembly through scaffolding. Scaffolding is the process of joining contigs into longer sequences (Venter et al. 1998), commonly referred to scaffolds. Scaffolding is performed using programs called scaffolders (Sahlin et al. 2014), but may also be performed internally by the *De novo* assembler (Simpson et al. 2009). The scaffolders use PE reads (Sahlin et al. 2014), or long-reads (Warren et al. 2015) to determine which contigs are most likely to originate from closely located sequences, and to

determine the order of such contigs. Typically, the reads (long-reads, or PEs) are aligned to the previously assembled contigs. Thereafter, a graph, or table is constructed to specify the distance between contigs (Sahlin et al. 2014; Warren et al. 2015). Simplifying such graphs, the order, and closeness of contigs may be determined, and the result is returned as a Fasta file, containing the scaffolds.

Lastly, the scaffolds may be joined, and validated through complementary methods, including FISH (Shearer et al. 2014), and Sanger sequencing (Goldberg et al. 2006).

Scaffolds may also be evaluated, and joined through comparative genomics (Mikkelsen et al. 2005; Zimin et al. 2009) .

1.7.3 Quality control and filtering

Once the sequencing data is assembled or aligned, preprocessing of the aligned data takes place. Preprocessing includes filtering (<http://broadinstitute.github.io/picard/>), QC (Ewels et al. 2016), as well as general preprocessing, including indexing and sorting of the data (H. Li et al. 2009).

There is a diversity of filters which may be applied to the sequencing data. Which filter to apply depends partly on the previous assembly strategy (*i.e. de novo* or mapping), and the downstream analysis of the WGS data. Briefly, the filtering of *de novo*-assemblies may include removal of short or poorly supported contigs, as well as removal of contaminant contigs (*i.e.* contigs not originating from the species/individual of interest, often originating from viruses or bacteria) (Nederbragt et al. 2010). Instead, the filtering of mapping assemblies includes removal of duplicates: reads that correspond to the exact same sequence; duplicates may either be marked (*i.e.* kept, but flagged as duplicate), or filtered (that is, removed from the dataset) (<http://broadinstitute.github.io/picard/>), other filters may include removal of contaminants, unaligned reads, or adapter sequences (<http://broadinstitute.github.io/picard/>).

Once the filtering is complete, QC is initiated. QC is needed to understand if, and to what degree the data can be trusted. Poor quality data may yield false positives, as well as false negatives. Common QC metrics include duplication rate (the percentage of reads flagged as duplicates), mean coverage, horizontal coverage (usually described as a percentage of the genome fulfilling a coverage threshold, such as 20X), the percentage of sequenced bases over a certain probability threshold, as well as the GC distribution of the sequenced reads (Ewels et al. 2016). Comparing these metrics to known high quality datasets, the user may infer if a newly sequenced dataset is of high or low quality.

Lastly, the more general pre-processing includes indexing and sorting; the reads are generally sorted based on their genomic coordinate, this is necessary to make the downstream analysis efficient (McKenna et al. 2010). Commonly, the reads are sorted using tools such as samtools (H. Li et al. 2009) and bedtools (Quinlan and Hall 2010). Once the preprocessing is complete, the downstream analysis is initiated. Downstream analyses vary depending on the organism and the purpose of the experiment. Downstream analysis typically involves variant detection, or comparative analysis between species or individuals (McKenna et al. 2010; Ekblom and Wolf 2014). The subsequent section will discuss the detection of SV in great detail.

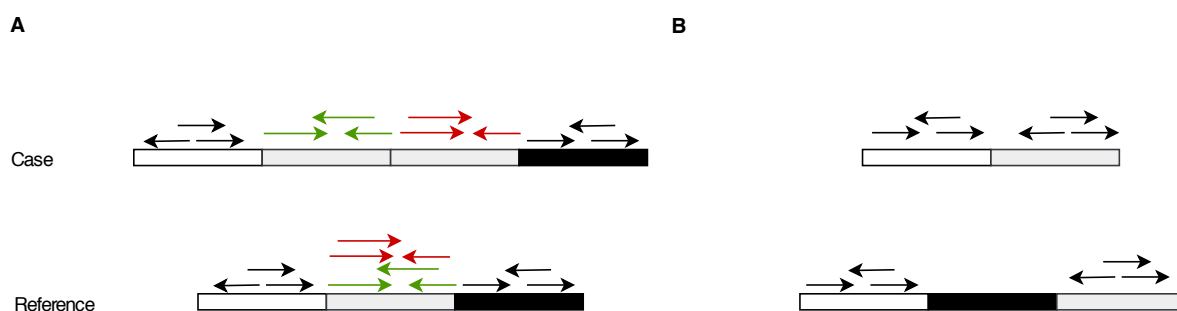
1.7.4 Structural variation calling

Once the preprocessing of the WGS data is complete, SV detection may be initiated. SV detection is performed by software dubbed “callers”; these software packages search the genome for signals indicating large genomic differences between the reference and the sequenced individual. In short-read sequencing data, these signals include unexpected regional read depth, split reads, as well as read pairs mapping in an unexpected pattern (Hormozdiari et al. 2009). Similarly, regional read depth, and split reads are used for the detection of SV in long-read sequencing data (Deschamps et al. 2018); however, since long-reads are unpaired, no read pair analysis can be performed.

In addition to the search and clustering of these signals, most callers include various filters to distinguish true variation from noise such as misalignments and contamination (Ye et al. 2009; Chen et al. 2015); some examples of these filters are found in **Paper I**.

Patterns in the read depth may be used to detect CNV. There is a large number of tools available for detecting CNV based on read depth signals; allowing detection of germline variants (Abyzov et al. 2011) as well as tumor normal analysis (Boeva et al. 2012). All read depth callers classify high coverage regions as duplications (Figure 7A), and low coverage regions as deletions (Figure 7B) (Abyzov et al. 2011; Boeva et al. 2012).

Figure 7. An illustration on how the read coverage depends on the copy number of a genomic region. The arrows indicate the WGS reads, and the colored boxes symbolize different genomic regions. A) A tandem duplication of the grey region, and B) a deletion of the black region.

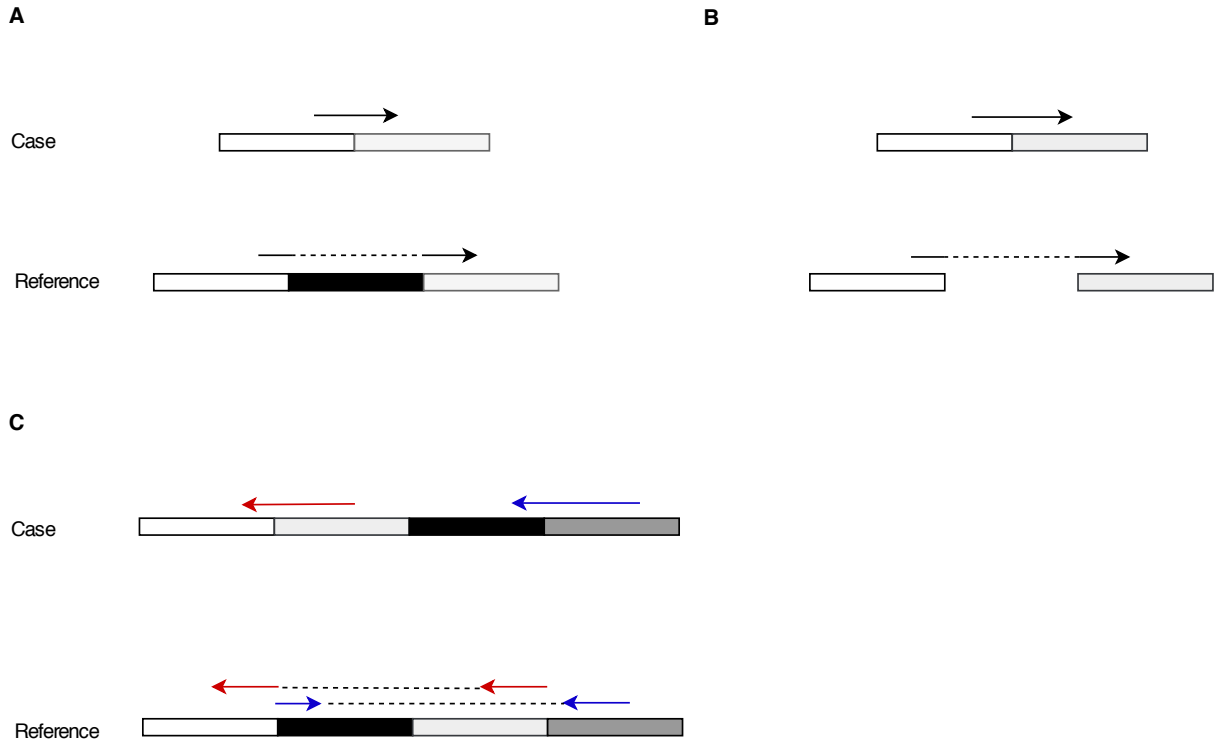


Usually, the callers apply GC normalization, as well as correction for low mappability. The detection of CNV usually involves segmentation. The aim of the segmentation process is to divide the genome into segments (regions) based on the local copy number. The copy number is inferred from the normalized read coverage, as well as the user given, or assumed ploidy of the organism. Once the genome is segmented into various copy number regions, the regions are quality checked and reported to the user (commonly via a VCF file) (Abyzov et al. 2011; Boeva et al. 2012).

Split reads are another commonly used signal used for SV detection. Split reads are defined as reads that are split across the reference (*i.e.* one part of the reads map to a certain region,

and the other part to a distant genomic region). Split reads may be used to detect a wide range of SVs, including CNVs, inversions, and translocations (Ye et al. 2009). These variants are recognized based on the orientation and positioning of the split reads. For instance, a deletion may be found by searching for split reads spanning the breakpoint junction (Figure 8A). Interchromosomal translocations are indicated by reads being split between two chromosomes (Figure 8B). An inversion may be recognized as split reads where half of the read maps to the forward strand, while the other half maps to the reverse strand of a distant region on the same chromosome (Figure 8C).

Figure 8. An illustration of how split reads signals arises in the WGS data. The arrows illustrate reads, arrows separated by dashed lines indicate split reads, and the colored boxes symbolize different genomic regions. A) A deletion of the black region. B) An interchromosomal translocation between the grey and white chromosome. C) Inversion of the grey and black regions.

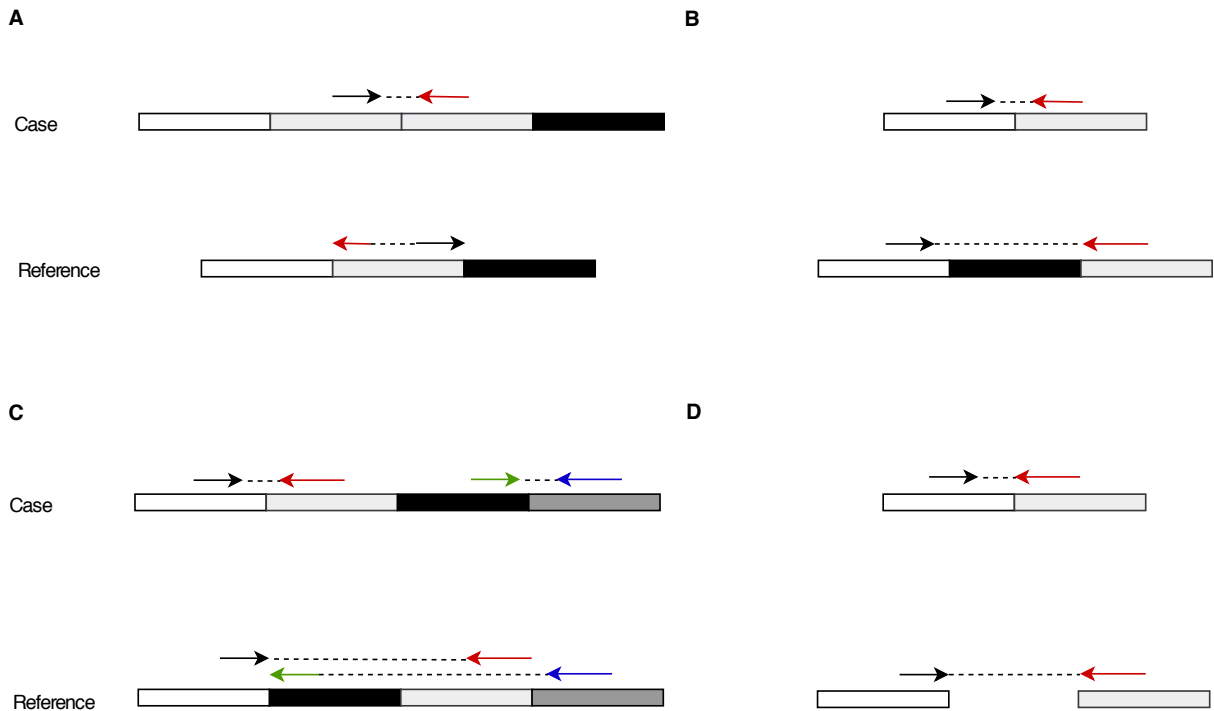


In contrast to the two other signals, split reads resolve the breakpoint junctions to the nucleotide level; which is useful to study the mechanisms of the formation of SV. Split read SV detection dominates the SV analysis in long-read sequencing data (Cretu Stancu et al. 2017); mainly because the reads are long enough to span the breakpoint junction as well as nearby repeats where SV often occur. Therefore long-reads are likely to map confidently to both sides of the breakpoint junction (Figure 8). In contrast the amount of split reads are limited in short-read sequencing data: the split reads are too short to span repeat regions, and significant parts of the read (usually >20 bp) must be located on both sides of the junction to produce confident alignments, limiting the detection rate of split read callers of short-read data (Tattini et al. 2015).

There are multiple split read callers available, such as Pindel, which uses a pattern growth algorithm to detect small SV using short-read sequencing data (usually less than 10Kbp) (Ye et al. 2009). Sniffles (Sedlazeck et al. 2018) and NanoSV (Cretu Stancu et al. 2017) are popular tools used for SV analysis in long-read WGS data.

Discordant pairs are the third signal that may be used to detect SV. Discordant pairs are read pairs that map in an unexpected pattern in relation to the other read pairs. Such pattern includes abnormal orientation and/or too large distance of the reads in a read pair (Tattini et al. 2015). Discordant pairs may be used to detect a wide range of variants, including CNVs, inversions and translocations (Figure 9). Discordant pairs effectively span repetitive or noisy regions of the genome, allowing detection of SV in these regions (this is especially true for large insert libraries). On the other hand, the insert size distribution is noisy, limiting the resolution of the discordant pair approach (Tattini et al. 2015).

Figure 9. An illustration of how discordant pairs arises in the WGS data. The pairs arrows illustrate the read pairs, and the colored boxes symbolize different genomic regions. As shown, the insert size is normal in the case, and extended to an abnormal length when aligned to the reference genome. The figure illustrates A) a tandem duplication of the grey region, B) A deletion of the black region, C) an inversion of the grey and black region, and D) a translocation between the gray and white chromosomes.



SV callers utilizing discordant pairs are among the most successful SV callers; they allow detection of a wide spectrum of sizes and types of variants, at a low computational cost. Hence, it is not surprising that a multitude of discordant pair callers has been developed (Chen et al. 2009; Tattini et al. 2015).

In addition to these three signals, SV may be detected through *de novo* assembly. There are two distinct *de novo* assembly approaches, local assembly and whole genome (WG) assembly (Baker 2012; Narzisi et al. 2013). WG *de novo* assembly aims to assemble all reads across the entire genome at once, thereby assembling the sequenced genome independently from the reference. Once the assembly process is complete, the resulting contigs are aligned to the reference genome – allowing a direct comparison between the patient and reference genome (Li 2015; Nattestad and Schatz 2016); once the contigs are

aligned to the genome, the SV may be found using an approach similar to split read SV detection (Figure 8). WG *de novo* assembly allows for detection and classification of all SV, and is the only effective method for detection of large novel sequence insertions and population specific sequence. However, the usage of WG *de novo* assembly is limited by high computational costs: WG *de novo* assembly of one individual may take weeks, and is a highly complicated process (Simpson et al. 2009). Despite these difficulties, there is a wide range of WG *de novo* assembly variant calling solutions available. Some of these are comprehensive and include all steps necessary to assemble and call variants (Li 2015). Other tools perform the variant calling only; these tools require the user to assemble and align the WGS data, requiring more knowledge while allowing for more flexibility (Nattestad and Schatz 2016).

In contrast to WG *de novo* assembly, local *de novo* assembly is a targeted approach where regions are extracted from the aligned sequencing data, and later re-assembled through *de novo* assembly. The local *de novo* assembly approach is usually faster than the WG approach (Narzisi et al. 2013). On the other hand, the results are highly dependent on the initial alignment of the data. Hence, local *de novo* assembly fails to capitalize on the greatest advantage of *de novo* assembly – a representation of the genome that is independent from the reference. Nevertheless, local *de novo* assembly algorithms are widely utilized for detection of small variants (usually less than 1Kbp) (Narzisi et al. 2013), or in concert with other signals (Chong et al. 2016).

Lastly, all, or any number of these signals may be combined. By combining multiple signals, the overall detection rate, precision, and SV classification is improved. Most of the current top performing callers use such an approach, including Manta (Chen et al. 2015) and TIDDIT (Eisfeldt et al. 2017). Manta detects SV through discordant pairs and split reads, and improves the classification of the candidate variants through local *de novo* assembly; this approach is particularly useful for small variants. Similarly, TIDDIT detects

variants through discordant pairs and split reads, but instead of local *de novo* assembly, TIDDIT analyses the read coverage to classify the variants; which is useful for analyzing large variants (>1Kbp).

1.8 ALGORITHM DEVELOPMENT

An algorithm is a specification of a procedure. Such procedure may have one, zero, or multiple inputs, as well as one or multiple outputs; these input signals are converted into the output signals through the procedure specified by the algorithm (Soare 2009). Algorithms are not confined to computer science, cooking recipes or construction plans may also be classified as algorithms. An algorithm is designed to solve a certain problem; it may be a complex problem, such as how to control a space shuttle, but may also be a simple problem, such as how to cook coffee. There are two classical approaches for designing algorithms: the top-down approach, or the bottom-up approach. In the top-down approach, the algorithm is detailed through decomposition: the designer starts from an overview of problem, and works inwards to specify the components in greater detail (Mostow 1985). Conversely, in the bottom-up approach, the designer creates chains of various simpler components, thereby constructing a more complex algorithm/software (Mostow 1985). Software algorithms can typically be created using either a top-down or a bottom-up approach.

Consider a simple pipeline, this pipeline accepts a BCL file as input, performs BCL to FASTQ conversion, alignment, variant calling, and annotation to produce an annotated VCF.

Using a top-down approach, the developer would create an overview of the system.

The input is BCL, processing takes place in between, and the output is a VCF file.

Next, the developer would decompose the processing block into submodules, such as preprocessing and variant analysis; these two blocks are then further decomposed, until the proposed pipeline is finished.

Instead, when using the bottom-up approach, the developer will piece various systems (software packages) together to form a new more complex system. The developer may start from the input, and decide to add the bcl2fastq tool (Illumina 2019), which takes a BCL file as input and produces a FASTQ file, thereafter, the developer adds an aligner, which inputs the resulting FASTQ and produces a BAM file. This process is continued until the entire pipeline is detailed.

Different algorithms may solve problems in different ways (in the same sense that many recipes may produce the same dish); such different approaches are referred to as algorithm paradigms. There are many algorithm paradigms, some are favored because they are simple to design, others because they may be turned into efficient software. Three common algorithm paradigms are brute force, greedy, and divide and conquer (Cormen et al. 2001). A brute force algorithm will search for the optimum results by testing all possible solutions of a problem (Cormen et al. 2001). For instance, an aligner may map a read to the reference by comparing the read to every position in the genome: such search would be slow, but in time, the aligner would find the best possible position of that read.

Greedy algorithms are designed to make the most rewarding choice at each given timepoint (*i.e.* the local optimum), by doing so, the software designer hopes to approach the best possible outcome (global optimum) (Cormen et al. 2001). Greedy algorithms are relatively easy and intuitive (*i.e.* cheap) to design, but in contrast to the brute force algorithm, the greedy algorithm will seldom reach the global optimum. An example of a greedy algorithm would be a chess player that always makes the most rewarding move; such strategy is unlikely to produce a victory (global optimum), but may initially provide good results.

Lastly, the divide and conquer algorithm involves splitting a problem into smaller problems. By dividing the problem into subsections, the problem may easily and quickly be solved (Cormen et al. 2001). Usually, these algorithms are recursive; the algorithm creates a smaller problem by dividing the input in two, these two parts are given as input to the

same algorithm, that once again divides the sub problems into two, until the problem is so small, that it is trivial to solve. Upon reaching this point, the algorithm reassembles the solutions of each sub-problem, until these solutions are assembled into the solution (output) of the initial, much greater and demanding problem (Hoare 1962). Such approach is usually time and memory efficient, especially on modern computers that involve a large number of processing cores (Cederman and Tsigas 2008); on the other hand, the divide and conquer approach is complicated to design and maintain, and may therefore be expensive to implement. Some of the most efficient sorting algorithms are classical examples of divide and conquer algorithms: both samtools (H. Li et al. 2009) and sambamba (Tarasov et al. 2015) utilize divide and conquer strategies to divide large WGS datasets into chunks that may quickly be sorted in the random access memory (RAM) of a computer.

A useful algorithm needs to be easy to maintain, and to perform its task using a limited amount of resources. The efficiency, or formally complexity, of an algorithm is usually described through the Big O (ordo) notation. The Big-O notation describe how the cost of running an algorithm is affected by the size of the input (Chivers and Sleightholme 2006). The complexity of an algorithm is usually given in the following form $O(x)$, where x is a mathematical function, such as n or $\log(n)$. For instance, an algorithm may have a linear time complexity (denoted $O(n)$), such algorithm will have a linear relationship between the time consumed and the size of the input. This could be the case for a software searching for words in a text file; if the length of the text is doubled, the software will need to search twice as many letters: resulting in a linear relationship between search time and text length. Some algorithms have a constant time complexity, the time consumption of such algorithm does not relate with the size of the input; this could be the case when searching the highest value in a list that is already sorted: the highest value will always appear at the top of the sorted list, no matter how many entries the list contains.

The sorting itself can be done at various time complexities: the worst sorting algorithm is arguably the WorstSort, an algorithm that is designed to never complete (Lerma 2014). The sort function of the Python language uses TimSort, having an average time complexity of $O(n \log(n))$ (<https://wiki.python.org/moin/HowTo/Sorting>).

The complexity of an algorithm may be determined theoretically or by running the algorithm with inputs of various sizes (Cormen et al. 2001).

1.9 PROGRAMMING LANGUAGES

Programming languages are the languages that programmers use to communicate with the computer. The Assembly languages, first developed in the late 1940s are among the earliest modern programming languages (Booth and Britten 1947). These languages are specific to a certain computer architecture, and are low level languages in the sense that there is a strong correspondence between the program statements and the machine code instructions (Sinkov et al. 1963). The need of more complex algorithms, greater portability, and lower cost of production has driven the development of a multitude of programming languages (Guarino 1978). Today there is a countless number of programming languages available. Some of these languages are designed to solve a specific task or run on a certain machine; such languages include Matlab (Mathworks Inc. 2016), which is designed for numerical computing, and the BASH language, which is mainly used in the Unix shell (Ramey et al. 2009). Other languages, such as Python (Rossum and Drake 1995) and Ruby (Flanagan and Matsumoto 2008) are designed to solve a wide variety of tasks, but are generally less optimized for any given problem. The following sections describes three programming languages: C++ (Stroustrup 1985), Python (Rossum and Drake 1995), and NextFlow (DI Tommaso et al. 2017). These three programming languages have been used extensively through all of the work presented in this thesis.

C++ is a general-purpose language, created by Bjarne Stroustrup in the early 1980s (Stroustrup 1985). C++ is to some degree an extension of C, and a significant amount of C code may be written within a functional C++ program. C++ is a compiled language, meaning that the source code is converted into machine code using a compiler. The compiler takes the source code as input, and outputs a file called binary, the binary file is subsequently executed by the user. Compared to interpreted languages, a compiled language offers faster runtime, and allows for the sharing of the compiled binary file (Plauser 2002). C++ is one of the most widely used languages, mainly because of its efficiency, portability, and stable community (Oualline et al. 1997). On the other hand, C++ is widely criticized for its overly complex syntax; leading to high production cost and low readability. As such, C++ is mainly preferred for large and complex software where control is necessary and a higher production cost may be justified; as well as for programs that perform heavy and complex computation, and therefore needs to be as efficient as possible.

Python is a general-purpose language released 1991 (Rossum and Drake 1995). Python is an interpreted language, meaning that the source code is typically not compiled into an executable file; instead the python source-code is read by an interpreter, a program that reads the source-code and executes it directly. Interpreted languages are generally easier to develop and debug compared to compiled languages, but are usually less efficient (Sanner 1999). Python is designed to be readable and easy to learn, it offers automatic memory management, dynamic-typing, and many other abstractions from the machine; further increasing its ease of use at the cost of efficiency (Rossum and Drake 1995).

Taken together, Python is a language that is easy to learn, code, and maintain; but is inefficient compared to other languages such as C++ (Prechelt 2003). Python is ideal for prototyping, simple scripting, and for developing tools that do not require efficiency (such as graphical user interfaces or wrappers).

Published in 2017, *NextFlow* is a relatively modern programming language. NextFlow is a language designed for creating and running software pipelines; and contains a large set of functions for that purpose, including interfaces to job schedulers and software environments (DI Tommaso et al. 2017). This stands in stark contrast to python and C++, which are general-purpose languages, lacking such specialized functionality. NextFlow was designed to target a common bioinformatics problem: tools and pipelines are commonly designed for a specific environment and purpose, and it's common for tools not to function in other environments or for other purposes. Lacking documentation, and poor programming practice are other factors limiting the use of bioinformatic software. Nextflow aims to solve these problems by offering a simple way of constructing and deploying pipelines, including their environments and configuration files necessary to set up the tools; as such, Nextflow provides a way of making bioinformatic research reproducible and more useful (DI Tommaso et al. 2017). However, Nextflow is not the only tool aimed for solving this problem, some notable competitors include SnakeMake (Köster and Rahmann 2012), which is a python package; as well as Luigi, which is built using Python, and maintained by Spotify (<https://github.com/spotify/luigi>).

2 AIMS OF THE THESIS

The overall goal of this thesis was to develop tools and to evaluate technologies for clinical detection of chromosomal aberrations using massively parallel sequencing.

The specific aims are:

- Development of a computational pipeline for characterization of structural chromosomal variants with WGS (**Papers I, II, and V**)
- Comparison of WGS strategies for the detection of structural chromosomal variants (**Paper II**)
- Characterization of complex chromosomal rearrangements (**Paper II, III, and IV**).

3 MATERIALS AND METHODS

3.1 COHORT

In **Paper I**, the tools TIDDIT and SVDB are presented. TIDDIT was validated using WGS data produced by the Genome in a bottle (GIAB) consortium (Zook et al. 2014), additionally, we use a subset of the thousand genome dataset (Altshuler et al. 2010) for validating SVDB.

All patients included in **Paper II** were recruited at the Clinical Genetics department, Karolinska University Hospital (Stockholm, Sweden). The study covered 3 individuals, in which conventional chromosome analysis had identified a complex chromosomal rearrangement. All three patients had been referred for chromosome analysis because of a clinical phenotype including neurocognitive deficit.

In **Paper III**, we reported 21 individuals carrying clustered CNV. Five of these individuals were recruited at the Kennedy Center (Rigshospitalet, Copenhagen, Denmark), two at Sahlgrenska University Hospital (Gothenburg, Sweden), one at Linköping University Hospital (Linköping, Sweden) and 13 at the Karolinska University Hospital (Stockholm, Sweden). All of these 21 individuals had previously been referred to one of these hospitals because of autism or intellectual disability.

In **Paper IV**, we perform *de novo* assembly of the WGS data of 1000 Swedish individuals; the WGS data was produced and made publicly available through the SweGen project. The SweGen cohort represent a cross section of the Swedish population, and consists of individuals drafted from the Swedish twin registry and Northern Sweden Population Health Study (Ameur et al. 2017).

All patients included in **Paper V** were recruited at the Clinical Genetics department, Karolinska University Hospital (Stockholm, Sweden). In total we report 324 individuals in 3 cohorts. Cohort 1 (The validation cohort) consisted of 68 individuals harboring three trisomies and 79 CNVs previously detected by Array comparative hybridization (aCGH) or

multiplex ligation-dependent probe amplification. Cohort 2 (The monogenic disease study cohort) consisted of 156 individuals referred for WGS with *in silico* gene panel analysis due to a clinical suspicion of monogenic disease within the areas of neuromuscular disorders, connective tissue disorders, unknown syndromes, skeletal dysplasias, hereditary cancer or other rare suspected mendelian conditions. Cohort 3 (The prospective study cohort) consisted of the unselected first 100 individuals referred for aCGH in 2017.

3.2 SHORT-READ SEQUENCING

PCR-free PE Illumina WGS was performed in **Paper II, III, and V**; The WGS was performed by the National Genomics Infrastructure (NGI) Stockholm (**Paper II, III, and V** cohort 1), as well as Clinical Genomics Stockholm facility (Science for Life Laboratory, Sweden) (**Paper V**, cohort 2, and 3). Across all studies, the coverage was roughly 30X, the insert size 350 bp, and read length 2x151 bp.

MP WGS was performed in **Paper II, and III**. These libraries were sequenced and prepped at the Kennedy Center (**Paper III**), or at NGI Stockholm (**Paper II**). In both papers, the libraries were sequenced on the Illumina HiSeq 2500 platform. The MP libraries were sequenced to an average depth of 3X, and the insert size was averagely 3Kbp (**Paper II**) or 6 Kbp (**Paper III**).

Linked-read sequencing was performed in **Paper II, and III**. The libraries were prepared by NGI Stockholm, using the 10X Chromium controller, the resulting libraries were sequenced to an average depth of 30X, using the Illumina HiSeq XTen platform.

All WGS data was delivered to the UPPMAX compute infrastructure, and analyzed using various custom pipelines detailed in the following sections. In **Paper V**, cohort 2 and 3 were analyzed on the Clinical Genomics Stockholm production high performance computing resource.

3.3 PREPROCESSING OF SHORT-READ WGS DATA

Preprocessing (*i.e.* adapter trimming, assembly, filtering, and QC) was performed using a variety of pipelines. In **Paper II**, **III**, and **V**, the Illumina short-read WGS data was preprocessed using the NGI-piper pipeline (<https://github.com/johandahlberg/piper>), which performs preprocessing and SNV calling according to the GATK best-practices for germline WGS data (McKenna et al. 2010). The resulting data is quality controlled using the MultiQC package (Ewels et al. 2016).

Cohort 2 and 3 of **Paper V** were sequenced at Clinical Genomics Stockholm, and were therefore preprocessed using the MIP pipeline (Stranneheim et al. 2014).

The Linked-read WGS data of **Paper II** and **Paper III** was preprocessed using the Longranger pipeline as well as the *Supernova de novo* assembler (Weisenfeld et al. 2017).

The contigs produced by supernova were aligned to hg19 using BWA MEM (Li 2013).

De novo assembly of the 1000 Swedish genomes was performed using the Assemblatron workflow, a workflow modelled after FermiKit (Li 2015) (**Paper IV**). The resulting contigs were aligned to hg19 and hg38 using BWA MEM, and the assemblies were quality controlled using the Assemblatron statistics module.

3.4 ANALYSIS OF BIONANO OPTICAL MAPS

Bionano optical maps were analyzed in **Paper II**. The optical maps were produced by BionanoGenomics (SanDiego,CA,USA), using the Saphyr platform (<https://bionanogenomics.com/support-page/saphyr-system>). The maps were detected using AutoDetect (version5.0svn:DM:r837), and assembled using the *de novo* assembly tool AssembleMolecules (version1.0). The resulting contigs were aligned to hg19 using the BionanoRefAligner (version5649). Finally, the results were visualized using Bionano Access, and the SMAP files were converted to vcf using the smap2vcf script (<https://github.com/J35P312/smap2vcf>).

3.5 SV ANALYSIS

In **Paper I**, SV calling was performed using Manta (Li 2015), Lumpy (Layer et al. 2014), CNVnator (Abyzov et al. 2011), Delly (Rausch et al. 2012), FermiKit (Li 2015), and TIDDIT (Eisfeldt et al. 2017).

In **Paper II, III, and V**, SV calling was performed using FindSV (<https://github.com/J35P312/FindSV>), a pipeline combining TIDDIT and CNVnator.

In **Paper IV**, novel sequences were found using samtools view (H. Li et al. 2009), selecting out all contigs that did not match the reference genome. The resulting contigs were clustered using CD-hit (Li and Godzik 2006), and analyzed using a variety of tools, including BLAST (Altschul et al. 1990) and NUCmer (Marçais et al. 2018).

3.6 SNV ANALYSIS

SNV analysis was performed in **Paper V**. SNVs were called using MIP version 6.0, a pipeline (Stranneheim et al. 2014) that performs SNV calling using the GATK haplotype caller (McKenna et al. 2010), samtools mpileup (H. Li et al. 2009), and Freebayes (Garrison and Marth 2012). The resulting VCF files are combined using GATK combine variants, and annotated using SNPEff (Cingolani et al. 2012), VEP (McLaren et al. 2016), and GENMOD (<https://github.com/moonso/genmod>). The resulting calls are ranked and sorted based on a variety of metrics, including population frequencies, inheritance patterns, and deleteriousness (Adzhubei et al. 2013; Kircher et al. 2014).

Lastly, the calls were filtered using the PanelApp (<https://panelapp.genomicsengland.co.uk/>) intellectual disability gene panel, as well as custom human phenotype ontology (HPO) terms (Köhler et al. 2019).

3.7 SOFTWARE AND PIPELINES

Various scripts and computer algorithms were developed in all papers (**Paper I, II, III, IV, V**). All of these tools are available on Github (<https://github.com/J35P312>). The following section describes a subset of such software tools.

3.7.1 TIDDIT and SVDB

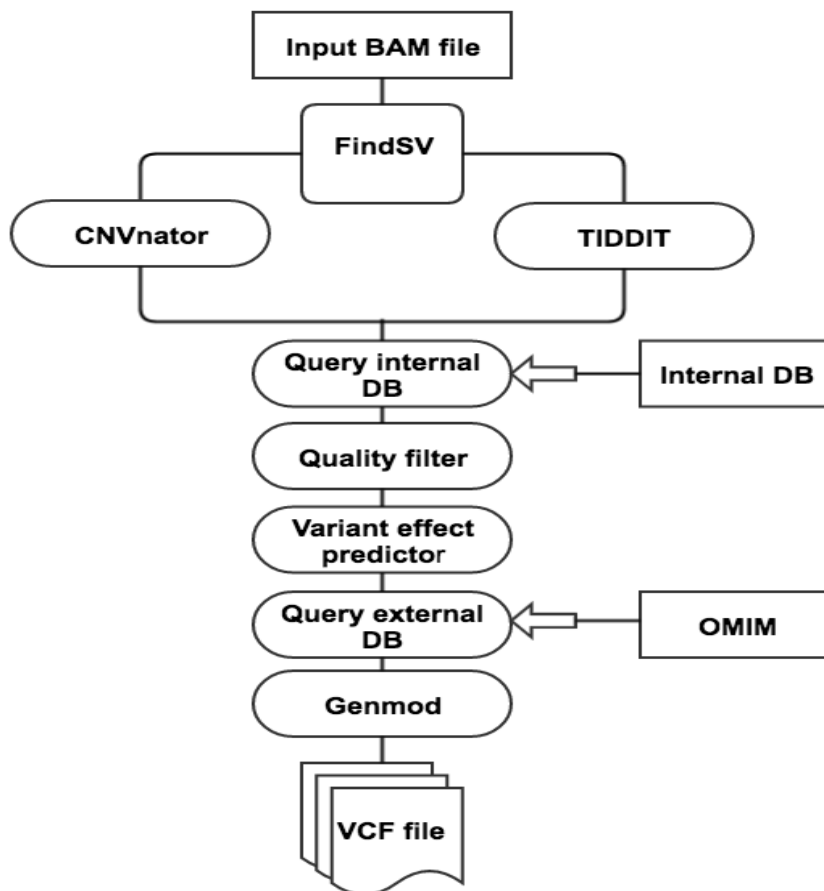
TIDDIT and SVDB were developed in **Paper I**. SVDB is a tool for creating SV frequency databases, and for merging and comparing SV callsets, TIDDIT is an SV caller, including basic QC and coverage analyses. SVDB was written in Python, and TIDDIT was written in C++. At the core, both of these tools use similar custom implementations of density-based spatial clustering of applications with noise (DBSCAN) (Ester, M., Kriegel, H. P., Sander, J., & Xu 1996). The software differs in that SVDB cluster calls, while TIDDIT cluster SV signals (supplementary alignments and discordant pairs). Both software outputs VCF files, but SVDB uses VCF as input, instead TIDDIT analyzes BAM files.

The tools are independent, and SVDB may be used to analyze the calls of most of the major SV callers (including Manta, Delly, CNVnator, FermiKit, Longranger, MELT (Gardner et al. 2017), and ExpansionHunter (Dolzhenko et al. 2017). SVDB is available through pip, PyPi, Github, and Bioconda (Dale et al. 2018), TIDDIT is available on Github, Bioconda, as well as a Singularity collection (Kurtzer et al. 2017).

3.7.2 FindSV

FindSV is an SV analysis pipeline, that performs SV calling and annotation using a BAM file as input (Figure 10). FindSV is implemented in NextFlow, but is run using a Python wrapper. FindSV is distributed using a Singularity collection. FindSV performs SV calling using CNVnator and TIDDIT. SVDB merges the calls produced by these callers, and the resulting VCF is annotated using VEP. Using SVDB, the resulting annotated VCF is annotated and ranked according to frequency; additionally, the tool Annotator is used to apply gene specific annotations based on the VEP annotation; Lastly, the VCF is ranked using GENMOD. FindSV was developed during the writing of **Paper I**. The resulting calls are commonly filtered according to size, frequency, or gene list.

Figure 10. A flowchart illustrating the FindSV pipeline. The FindSV pipeline performs SV calling using two complementary callers (TIDDIT and CNVnator), and includes multiple steps for variant filtering and annotation.



3.8 STATISTICAL ANALYSES

Statistical analyzes were performed in **Paper I, III, IV**. Nonparametric resampling tests were performed in **Paper III** and **Paper IV**; these tests were implemented in python, using the Numpy package (Oliphant and Millma 2006). In all cases, random subsets (*i.e.* sets of regions) were samples across the entire genome, and compared with the observations (*i.e.* regions of interests, such as SV breakpoints). A P value was obtained by calculating the fraction of times that the randomly selected regions were more extreme than the observed regions. Such tests were used to evaluate the enrichment of novel sequence within genetic regions such as genes and repeat elements in **Paper IV**, as well as to evaluate the enrichment of breakpoint microhomology, repeat elements and SNV in **Paper III**. TIDDIT performs a variety of statistical tests, in particular, it performs clustering using DBSCAN, and evaluates the likelihood of SV using a model similar to the scaffolder BESST (Sahlin et al. 2014) (**Paper I**). SVDB (Eisfeldt et al. 2017) performs clustering using DBSCAN and the Jaccard index(Jaccard 1901) (**Paper I**). ANOVA was used in **Paper IV** to test for differences between thousand genomes populations, and binomial tests were used to test for enrichments of various genetic features close to novel sequence insertions. Linear regression and Spearman correlation were used to evaluate the correlation between the Swedish novel sequences and Pan-African novel sequences (Sherman et al. 2019) in **Paper IV**.

3.9 MOLECULAR ANALYSES

Molecular analyses were performed in **Paper II, III, and V**. These analyses include aCGH, FISH, and karyotyping.

3.9.1 Array comparative hybridization (aCGH)

Genomic DNA was isolated from whole blood using standardized protocols and used for aCGH analysis. Three array designs were used: 1x180K custom oligonucleotide microarray, medical exome 1x1M Agilent oligonucleotide microarray, and a custom designed 2x400K array. The log₂ ratios were plotted and segmented by circular binary segmentation using the CytoSure Interpret software v4.10 (Oxford Gene Technology, Oxfordshire, UK). All CNVs were classified according to the American College of Medical Genetics (ACMG) guidelines.

3.9.2 Fluorescence in situ hybridization (FISH)

Fluorescence in situ hybridization (FISH) was performed using standardized protocols from peripheral blood cultures.

3.9.3 Karyotyping

Chromosome analysis was performed on metaphases from peripheral blood cultures according to standard protocols with subsequent G-banding with an approximate resolution of 550 bands per haploid genome. A minimum of 10 metaphases were analyzed.

4 RESULTS

SVs are known to contribute to the phenotypic diversity and disease traits of human individuals, and are therefore of interest in multiple fields, including rare diseases research and clinical diagnostics. Through these studies we test WGS methods, investigate SV formation mechanisms, and develop the SV analysis pipeline FindSV.

4.1 FindSV

FindSV is a computational pipeline that performs SV calling, filtering and annotation using Illumina WGS data. FindSV consists of public tools, including VEP, and CNVnator, as well as in-house developed tools; including the SV caller TIDDT and the SV database tool SVDB (**Paper I**). The pipeline was validated on roughly 100 CNVs detected using conventional cytogenetic methods, and was shown to perform favorably compared to existing SV analysis tools (**Paper I, Paper V**). We show that Illumina short read WGS is useful for finding and characterizing SV, and utilize our pipeline to detect disease causing SV and to study the mechanisms underlying SV formation (**Paper II, III, and IV**). In particular, we study complex rearrangements (**Paper III**), as well as novel sequence insertions (**Paper IV**).

4.2 COMPLEX GENOMIC REARRANGEMENTS

In **Paper III**, we study 21 complex CNVs, and find that complex CNVs are formed through a diversity of mechanisms, including chromothripsis and chromoanasythesis.

4.3 COMPARISON OF WGS TECHNOLOGIES

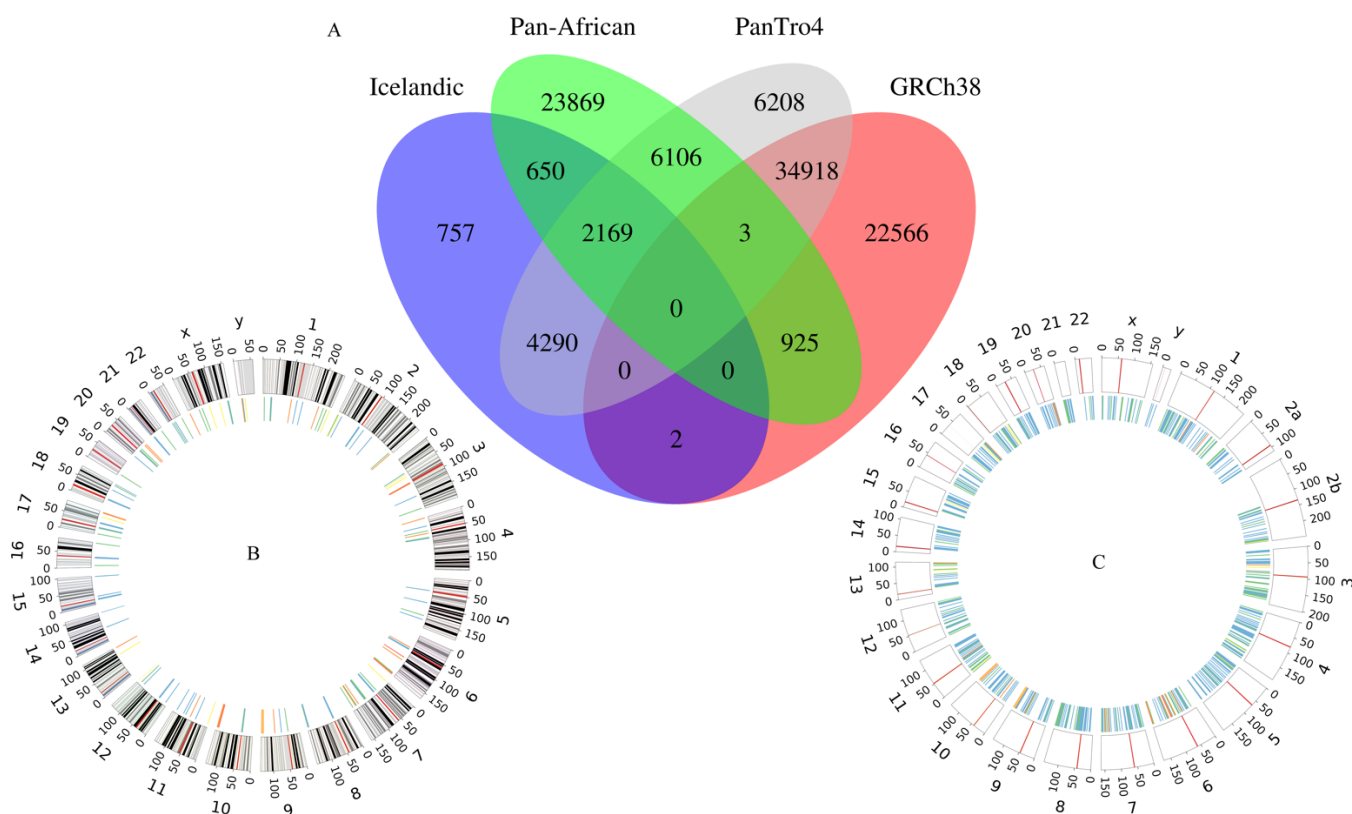
Additionally, we compare multiple sequencing methods, showing that they have specific advantages and disadvantages, and that the optimum approach for SV detection would include multiple complementary methods (**Paper II**). For instance, we show that Bionano optical maps are suitable for spanning repetitive regions, while Illumina PCR-free WGS offers high resolution.

4.4 NOVEL SEQUENCES IN THE SWEDISH POPULATION

In **Paper IV**, we find that Swedish individuals carry an abundance of sequence not present in the human reference genome, most of which is of ancestral origin (Figure 11).

Interestingly, we find a diversity of non-reference sequence within known disease genes, indicating that novel sequence may act as risk alleles.

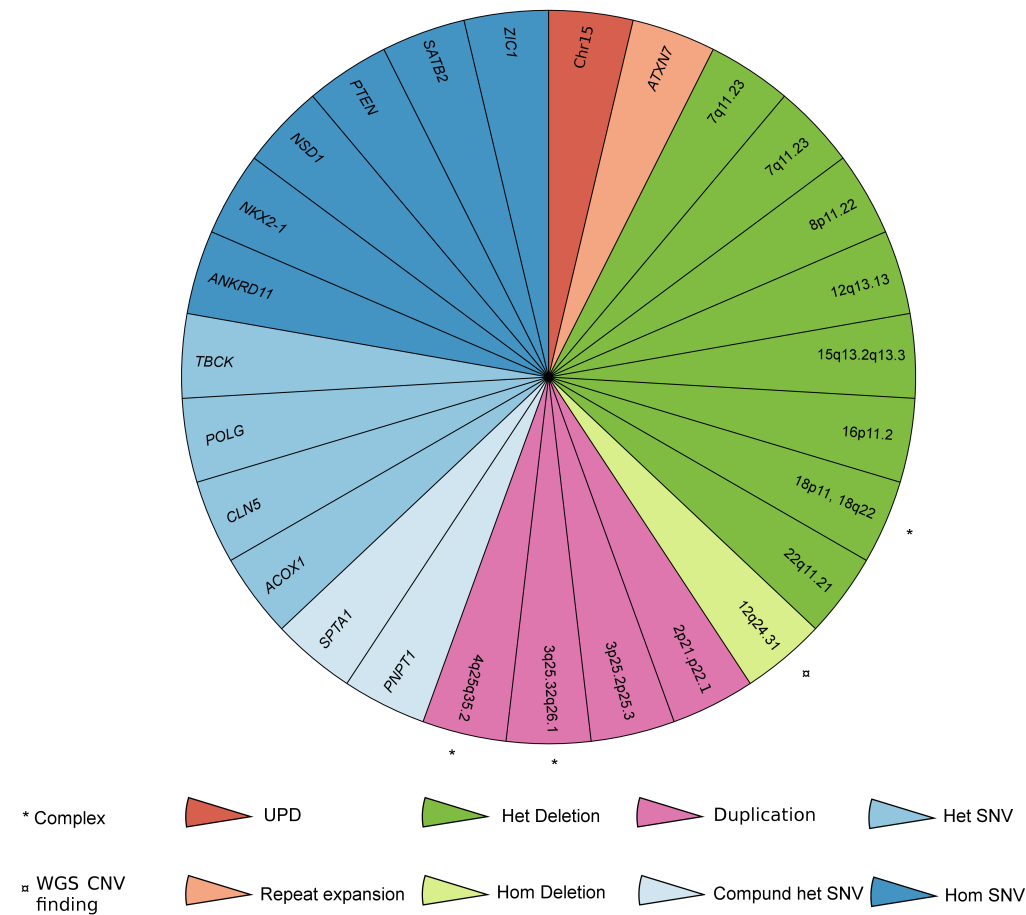
Figure 11. The Origin of sequences not present in hg19. A) The number of novel sequences mapping to public datasets. B, C) the distributions of contigs across hg38 and PT4. The colors of the inner circle indicate the percentage density of contigs within 1Mbp sized bins (white=0%, blue < 0.1%, green < 0.5%, yellow < 1%, orange < 10%, red < 20%, magenta < 40%, 40% > black). The figure is adapted from **Paper IV**.



4.5 AS A FIRST-TIER CLINICAL TEST IN GENETIC DIAGNOSTICS

Lastly, we validate the use of clinical WGS SV detection, and show that the diagnostic yield will be improved by using WGS as a first-tier test. In particular, we find that Illumina WGS is a comprehensive tool, allowing for the detection of SNV, SV, repeat expansions, and uniparental disomy (UPD) in a single experiment (**Paper V**) (Figure 12).

Figure 12. A pie chart illustrating clinically relevant findings in the prospective cohort of Paper V. Each slice of the pie chart represents one individual in cases analyzed by both aCGH and WGS.



5 FUTURE PERSPECTIVES

Short-read Illumina WGS is becoming a first-tier diagnostic test; enabling the analysis of multiple types of variation in a single experiment. Although short-read WGS is a powerful and comprehensive test, the diagnostic yield is still relatively low (Figure 12). A systems biology approach will be needed to improve the clinical assessment of genetic variation, as well as to increase our knowledge of the human genome. In particular, transcriptome and proteome analyses will be needed for understanding disease causing mutations within non-coding regions, as well as to evaluate variants of unknown significance. Therefore, the field of medical genetics is likely to broaden, and include a wider range of tests for predicting the health state of the patients. As the cost of large-scale molecular testing decreases, these tests may be applied during routine check-ups; offering early detection of cancer, as well as personalized medicine.

Similar to the great advance in the genetic field, smartphones and other mobile devices are becoming increasingly powerful, and are used for measuring a variety of health-related metrics in real-time. Such health-metrics include sleep patterns, social and physical activity, as well as heart rate. These metrics may be of great use in healthcare, and is likely to be combined with the molecular tests; perhaps using machine learning and artificial intelligence. The combination of molecular tests and high-resolution behavioral patterns will be of great use for assessing the health state of individuals: predicting disease and identifying risk behavior; in the near future, we will know ourselves better than ever before.

6 ACKNOWLEDGEMENTS

I want to thank everyone taking part in this great and exciting adventure! I want to begin by thanking my six excellent supervisors:

Anna Lindstrand, I thank you for your humor, kindness, and generosity, for all the great data, travels and computers. I also thank you for the freedom of leading my own projects; together, we have explored all types of variants, the chimpanzee genome, tested novel technologies, and much more. I look forward to continuing our work in the clinic!

Daniel Nilsson, for all our interesting discussions, ranging from bioinformatics, to topics such as books, philosophy, and ancient technology. It has been great to have you as a co-supervisor and neighbor at clinical genomics!

Henrik Stranneheim, thanks for joining our team, and for bringing order to the code, as well as your friendliness and kindness. It will be great fun to continue and work together.

Francesco Vezzi, for all the help with programming and designing TIDDIT and SVDB; for all lunches, writing sessions, and after-work meetings. It has been great fun to work with you!

Magnus Nordenskjöld, for your great knowledge and wisdom! For your support, and for teaching me how to ride the kick-scooter.

Valtteri Wirta, for all the great group retreats and meetings, for teaching me the true way of conferencing, as well as your great humor; I also want to thank you for my beautiful desk, with great view on the trees, birds, and hardworking lab-people.

My Mentor **Pall Olason**, who has left us for a grayer and rainier place; thank you for your help in benchmarking the SV callers, as well as all the fun visiting Uppsala. Hope Iceland and Decode treats you well!

I also want to thank everyone at **The Clinical genomics facility**, for being such a welcoming and happy team; many thanks for accepting me as a part of it; **Adam Rosenbaum, Anna Engström, Anna Zetterlund, Barry Stockman, Chiara Rasi, Emilia Ottosson Laakso, Emma Sernstad, Karin Sollander, Lars Engstrand, Maya Brandi, Michael Akhras, Mikael Laaksonen, Moa Hägglund, Patrik Grenfeldt, and Sarath Murugan**. Especially, I want to thank **Anders Jemt, Måns Magnusson, Hassan Foroughi, Tanja Normark, and Isak Sylvén** for bioinformatic collaboration and discussions. To **Daniel Backman** for arranging really fun gaming evenings; and **Anna Leinfeldt** for helping me and my students getting in and out of SciLifeLab; **Kenny Billiau**, for letting me in to Rasta. Last, but certainly, not least I want to thank our great Party Generals **Keyvan Elhami, Cecilia Svensson, and Sofie Sibia** for arranging parties, After-works, and Karaoke sessions.

All collaborators at NGI, including **Max Käller, Phil Ewels, Maxime Garcia, Szilvester Juhos, Remi Olssen, Mattias Ormestad, and Per Lundin**. Thank you for providing us with great data, and for all discussions, seminars, collaborations, and for keeping me company throughout various user-group meetings and conferences.

I thank everyone in the rare disease and clinical genetics groups; **Ann Nordgren, Benedicte Bang, Bianca Tesi, Charlotte Willfors, Ellika Sahlin, Josephine Wincent, Lisselotte Vesterlund, Emeli Ponten, Dominyka Batkovskytė, Giedre Grigelionienė, Karin Salehi, Malin Kvarnå, Mansoureh Shahsavani, Sintia Kolbjørn, Sofia Frisk, Vasilios Zachariadis and Wolfgang Hoffmeister**. Especially, I thank, **Nina Jäntti** for your humor and kindness; **Maria Pettersson** for your enthusiasm, and for the many and fun collaborations that we have had. I thank **Raquel Vaz, Anna Hammarsjö, Jacob Schuy,**

for their friendliness. Many thanks to **Fulya Taylan** for being helpful, and for our many and great discussions.

Bioclinicum floor 10 for all the projects, after works, and company! We are truly at the top of Karolinska; **Abbe Ullgren, Anaya Mukherjee, Agneta Nordensköld, Anders Kämpe, Aron Skaftason, Behzad Khoshnood, Clara Ibel, Cecilia Östholm Corbascio, Carro Graff, Daniel Hägerstrand, Emma Ehn, Isabel Tapia Paez, Jessica Alm, Jose Laffita, Kali Patra, Larry Mansouri, Lesley-Ann Sutton, Mikaela Friedman, Richard Rosenquist Brandell, Samina Asad.** In particular, I thank **Karthick Natarajan**, for his great humor and wisdom! **Kelda Stagg**, for arranging such fun after works, and for bringing color (especially blue), to our floor.

Jessada Thutkawkorapin, for your friendliness, as well as our bioinformatic discussions. **Alice Constantini** for always being friendly and happy!

All co-workers at **Klinisk genetik KS**, who have provided me with interesting data, projects and great company throughout all meetings and conferences. Especially, I want to thank, **Agne Lieden, Britt Marie Anderlid, Cecilia Arthur, Christa Costa, Diego Cortese, Elisabeth Syk Lundberg, Erik Ivarsson, Emma Tham, Gisela Barbany, Helena Malmgren, Hero Nikdin Awier, Håkan Thornberg, Ingegerd Ivanov Öfverholm, Karin Wallander, Kristina Lagerstedt, Johanna Lundin, Mia Soller, Fatemah Rezayee, Nadja Pekkola-Pacheco. Ahn Nhi Tran** for your great happiness and friendliness.

The **MMK administration**, especially, **Ann-Britt Wikström** for being quick and friendly in sorting out the great amount of paperwork, **Jan-Erik Karre** for arranging our local backup server.

UPPMAX and **SNIC** for granting me the vast amount of computational resources needed for completing these projects. Together, we have burnt roughly 3 million core hours! In particular I want to thank **Marcus Lundberg** and **Valentin Georgiev** for being generous with core hours and disk space.

My project students, **Vanja Börjesson**, **Rasmus Larsson**, and **Alexander Kvist**, good luck to all of you!

All collaborators taking part in this and related work; including **Adam Ameer**, **Anna Petri**, **Björn Nystedt**, **Charlotte Gran**, **Claudia Carvalho**, **Hans Mattsson**, **Johanna Andersson-Assarsson**, **Lars Feuk**, **Lucia Pena**, **Lusine Nazaryan**, **Nikolas Harold**, **Robert Månsson**, **Sara Dahl**, and **Zeynep Tumer**.

I want to thank all of my friends; especially, I want to thank **Jonas Juhlin**, “**Seb**”, **Elin Parsjö**, and **Matilda**, as well as all my friends and course mates and friends from **Umeå**, in particular, I want to thank all of **Bit10 UMU**: “*Deus creat, nos mutamus*”.

All my **family**.

Lastly, I want to thank my precious roommates in my small Södermalm shoebox: **Simon** the Spider, **Malin** the Moth, and **Sigrid** the Silverfish, for keeping me company.

7 REFERENCES

- Abyzov A, Urban AE, Snyder M, Gerstein M. 2011. CNVnator: An approach to discover, genotype, and characterize typical and atypical CNVs from family and population genome sequencing. *Genome Res.* 21:974–984.
- Adzhubei I, Jordan DM, Sunyaev SR. 2013. Predicting functional effect of human missense mutations using PolyPhen-2. *Curr. Protoc. Hum. Genet.* 76:7–20.
- Albert TJ, Molla MN, Muzny DM, Nazareth L, Wheeler D, Song X, Richmond TA, Middle CM, Rodesch MJ, Packard CJ, et al. 2007. Direct selection of human genomic loci by microarray hybridization. *Nat. Methods* 4:903–905.
- Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. 1990. Basic local alignment search tool. *J. Mol. Biol.* 215:403–410.
- Altshuler DL, Durbin RM, Abecasis GR, Bentley DR, Chakravarti A, Clark AG, Collins FS, De La Vega FM, Donnelly P, Egholm M, et al. 2010. A map of human genome variation from population-scale sequencing. *Nature* 467:1061–1073.
- Ameur A, Dahlberg J, Olason P, Vezzi F, Karlsson R, Martin M, Viklund J, Kähäri AK, Lundin P, Che H, et al. 2017. SweGen: A whole-genome data resource of genetic variability in a cross-section of the Swedish population. *Eur. J. Hum. Genet.* 25:1253–1260.
- Babu MM, Luscombe NM, Aravind L, Gerstein M, Teichmann SA. 2004. Structure and evolution of transcriptional regulatory networks. *Curr. Opin. Struct. Biol.* 14:283–291.
- Baca SC, Prandi D, Lawrence MS, Mosquera JM, Romanel A, Drier Y, Park K, Kitabayashi N, MacDonald TY, Ghandi M, et al. 2013. Punctuated evolution of prostate cancer genomes. *Cell* 153:666–677.
- Baker M. 2012. De novo genome assembly: What every biologist should know. *Nat. Methods* 9:333–337.
- Balasubramanian S. 2011. Sequencing nucleic acids: From chemistry to medicine. *Chem. Commun.* 47:7281–7286.
- Barba M, Czosnek H, Hadidi A. 2013. Historical perspective, development and applications of next-generation sequencing in plant virology. *Viruses* 1:106–136.
- Barski A, Cuddapah S, Cui K, Roh TY, Schones DE, Wang Z, Wei G, Chepelev I, Zhao K. 2007. High-Resolution Profiling of Histone Methylations in the Human Genome. *Cell* 129:823–837.
- Bell DC, Thomas WK, Murtagh KM, Dionne CA, Graham AC, Anderson JE, Glove WR. 2012. DNA base identification by electron microscopy. *Microsc. Microanal.* 18:1049–1053.
- Bennett S. 2004. Solexa Ltd. *Pharmacogenomics* 5:433–438.
- Boeva V, Popova T, Bleakley K, Chiche P, Cappo J, Schleiermacher G, Janoueix-Lerosey I, Delattre O, Barillot E. 2012. Control-FREEC: A tool for assessing copy number and allelic content using next-generation sequencing data. *Bioinformatics* 28:423–425.
- Bolser DM, Staines DM, Perry E, Kersey PJ. 2017. Ensembl plants: Integrating tools for

- visualizing, mining, and analyzing plant genomic data. In: *Methods in Molecular Biology*. p. 1–31.
- Boomsma DI, Wijmenga C, Slagboom EP, Swertz MA, Karssen LC, Abdellaoui A, Ye K, Guryev V, Vermaat M, Van Dijk F, et al. 2014. The Genome of the Netherlands: Design, and project goals. *Eur. J. Hum. Genet.* 22:221–227.
- Booth AD, Britten KH. 1947. GENERAL CONSIDERATIONS IN THE DESIGN OF AN ALL PURPOSE ELECTRONIC DIGITAL COMPUTER.
- Bowling KM, Thompson ML, Amaral MD, Finnila CR, Hiatt SM, Engel KL, Cochran JN, Brothers KB, East KM, Gray DE, et al. 2017. Genomic diagnosis for children with intellectual disability and/or developmental delay. *Genome Med.* 30:43.
- Brenner S, Johnson M, Bridgham J, Golda G, Lloyd DH, Johnson D, Luo S, McCurdy S, Foy M, Ewan M, et al. 2000. Gene expression analysis by massively parallel signature sequencing (MPSS) on microbead arrays. *Nat. Biotechnol.* 18:630–634.
- Bronner IF, Quail MA, Turner DJ, Swerdlow H. 2014. Improved protocols for Illumina sequencing. *Curr. Protoc. Hum. Genet.* 79:18.2.1-18.2.42.
- Cao Hongzhi, Hastie AR, Cao D, Lam ET, Sun Y, Huang H, Liu X, Lin L, Andrews W, Chan S, et al. 2014. Rapid detection of structural variation in a human genome using nanochannel-based genome mapping technology. *Gigascience* 3:34.
- Cederman D, Tsigas P. 2008. A practical quicksort algorithm for graphics processors. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. p. 246–258.
- Chan EKF, Cameron DL, Petersen DC, Lyons RJ, Baldi BF, Papenfuss AT, Thomas DM, Hayes VM. 2018. Optical mapping reveals a higher level of genomic architecture of chained fusions in cancer. *Genome Res.* 28:726–738.
- Chen K, Wallis JW, McLellan MD, Larson DE, Kalicki JM, Pohl CS, McGrath SD, Wendl MC, Zhang Q, Locke DP, et al. 2009. BreakDancer: An algorithm for high-resolution mapping of genomic structural variation. *Nat. Methods* 6:677–681.
- Chen X, Schulz-Trieglaff O, Shaw R, Barnes B, Schlesinger F, Cox AJ, Kruglyak S, Saunders CT. 2015. Manta: Rapid detection of structural variants and indels for clinical sequencing applications. *bioRxiv [Internet]* 32:024232. Available from: <http://www.biorxiv.org/content/early/2015/08/10/024232.abstract>
- Chiang CWK, Mangul S, Robles C, Sankararaman S. 2018. A Comprehensive Map of Genetic Variation in the World's Largest Ethnic Group-Han Chinese. *Mol. Biol. Evol.* 35:2736–2750.
- Chivers ID, Sleightholme J. 2006. Introduction to Programming with Fortran with Civerage of Fortran 90,95,2003 and 77.
- Chong Z, Ruan J, Gao M, Zhou W, Chen T, Fan X, Ding L, Lee AY, Boutros P, Chen J, et al. 2016. NovoBreak: Local assembly for breakpoint detection in cancer genomes. *Nat. Methods* 14:65–67.
- Chopra R, Burow G, Farmer A, Mudge J, Simpson CE, Burow MD. 2014. Comparisons of de novo transcriptome assemblers in diploid and polyploid species using peanut (*Arachis spp.*) RNA-Seq data. *PLoS One* 10:e115055.

- Cingolani P, Platts A, Wang LL, Coon M, Nguyen T, Wang L, Land SJ, Lu X, Ruden DM. 2012. A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of *Drosophila melanogaster* strain w1118; iso-2; iso-3. *Fly (Austin)*. 6:80–92.
- Collins FS, Lander ES, Rogers J, Waterson RH. 2004. Finishing the euchromatic sequence of the human genome. *Nature* 431:931–945.
- Collins RL, Brand H, Karczewski KJ, Zhao X, Alföldi J, Khera A V., Francioli LC, Gauthier LD, Wang H, Watts NA, et al. 2019. An open resource of structural variation for medical and population genetics. *bioRxiv* [Internet]:578674. Available from: <https://www.biorxiv.org/content/10.1101/578674v1>
- Collins RL, Brand H, Redin CE, Hanscom C, Antolik C, Stone MR, Glessner JT, Mason T, Pregno G, Dorrani N, et al. 2017. Defining the diverse spectrum of inversions, complex structural variation, and chromothripsis in the morbid human genome. *Genome Biol.* 18:36.
- Cormen TH, Leiserson CE, Rivest RL. 2001. *Introduction to Algorithms*, Second Edition.
- Cretu Stancu M, Van Roosmalen MJ, Renkens I, Nieboer MM, Middelkamp S, De Ligt J, Pregno G, Giachino D, Mandrile G, Espejo Valle-Inclan J, et al. 2017. Mapping and phasing of structural variation in patient genomes using nanopore sequencing. *Nat. Commun.* 8:1326.
- Dale R, Grüning B, Sjödin A, Rowe J, Chapman BA, Tomkins-Tinch CH, Valieris R, Batut B, Caprez A, Cokelaer T, et al. 2018. Bioconda: Sustainable and comprehensive software distribution for the life sciences. *Nat. Methods* 15:475.
- Dekker J, Rippe K, Dekker M, Kleckner N. 2002. Capturing chromosome conformation. *Science* (80-.). 295:1306–1311.
- Depledge DP, Srinivas KP, Sadaoka T, Bready D, Mori Y, Placantonakis DG, Mohr I, Wilson AC. 2019. Direct RNA sequencing on nanopore arrays redefines the transcriptional complexity of a viral pathogen. *Nat. Commun.* 10:754.
- Deschamps S, Zhang Y, Llaca V, Ye L, Sanyal A, King M, May G, Lin H. 2018. A chromosome-scale assembly of the sorghum genome using nanopore sequencing and optical mapping. *Nat. Commun.* 9:4844.
- Dohm JC, Lottaz C, Borodina T, Himmelbauer H. 2008. Substantial biases in ultra-short read data sets from high-throughput DNA sequencing. *Nucleic Acids Res.* 36:e105.
- Dolzhenko E, van Vugt JJFA, Shaw RJ, Bekritsky MA, Van Blitterswijk M, Narzisi G, Ajay SS, Rajan V, Lajoie BR, Johnson NH, et al. 2017. Detection of long repeat expansions from PCR-free whole-genome sequence data. *Genome Res.* 27:1895–1903.
- van der Donk R, Jansen S, Schuurs-Hoeijmakers JHM, Koolen DA, Goltstein LCMJ, Hoischen A, Brunner HG, Kemmeren P, Nellåker C, Vissers LELM, et al. 2018. Next-generation phenotyping using computer vision algorithms in rare genomic neurodevelopmental disorders. *Genet. Med.* 21:1719-1725.
- Drmanac R, Sparks AB, Callow MJ, Halpern AL, Burns NL, Kermani BG, Carnevali P, Nazarenko I, Nilsen GB, Yeung G, et al. 2010. Human genome sequencing using unchained base reads on self-assembling DNA nanoarrays. *Science* (80-.). 327:78–81.

- Dunham I, Kundaje A, Aldred SF, Collins PJ, Davis CA, Doyle F, Epstein CB, Fietze S, Harrow J, Kaul R, et al. 2012. An integrated encyclopedia of DNA elements in the human genome. *Nature* 489:57–74.
- Ehrenhofer-Murray AE. 2004. Chromatin dynamics at DNA replication, transcription and repair. *Eur. J. Biochem.* 271:2335–2349.
- Eisfeldt J, Vezzi F, Olason P, Nilsson D, Lindstrand A. 2017. TIDDIT, an efficient and comprehensive structural variant caller for massive parallel sequencing data. *F1000Research* [Internet] 6:664. Available from: <https://f1000research.com/articles/6-664/v2>
- Ekblom R, Wolf JBW. 2014. A field guide to whole-genome sequencing, assembly and annotation. *Evol. Appl.* 7:1026–1042.
- English BP, Hauryliuk V, Sanamrad A, Tankov S, Dekker NH, Elf J. 2011. Single-molecule investigations of the stringent response machinery in living bacterial cells. *Proc. Natl. Acad. Sci. U. S. A.* 108:E365–E373.
- Ester, M., Kriegel, H. P., Sander, J., & Xu X. 1996. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. *Kdd* 96:226–231.
- Ewels P, Magnusson M, Lundin S, Käller M. 2016. MultiQC: Summarize analysis results for multiple tools and samples in a single report. *Bioinformatics* 32:3047–3048.
- Feuk L. 2010. Inversion variants in the human genome: Role in disease and genome architecture. *Genome Med.* 2:11.
- Flanagan D, Matsumoto Y. 2008. *The Ruby Programming Language*.
- Fullwood MJ, Wei CL, Liu ET, Ruan Y. 2009. Next-generation DNA sequencing of paired-end tags (PET) for transcriptome and genome analyses. *Genome Res.* 19:521–532.
- Gardner EJ, Lam VK, Harris DN, Chuang NT, Scott EC, Stephen Pittard W, Mills RE, Devine SE. 2017. The mobile element locator tool (MELT): Population-scale mobile element discovery and biology. *Genome Res.* 27:1916–1929.
- Garrison E, Marth G. 2012. Haplotype-based variant detection from short-read sequencing. *arXiv Prepr. arXiv1207.3907* 1207.
- Gigante S, Gouil Q, Lucattini A, Keniry A, Beck T, Tinning M, Gordon L, Woodruff C, Speed TP, Blewitt ME, et al. 2019. Using long-read sequencing to detect imprinted DNA methylation. *Nucleic Acids Res.* 47:e46–e46.
- Goldberg SMD, Johnson J, Busam D, Feldblyum T, Ferreira S, Friedman R, Halpern A, Khouri H, Kravitz SA, Lauro FM, et al. 2006. A Sanger/pyrosequencing hybrid approach for the generation of high-quality draft assemblies of marine microbial genomes. *Proc. Natl. Acad. Sci. U. S. A.* 103:11240–11245.
- Goldstein S, Beka L, Graf J, Klassen JL. 2019. Evaluation of strategies for the assembly of diverse bacterial genomes using MinION long-read sequencing. *BMC Genomics* 20:23.
- Grabherr MG, Haas BJ, Yassour M, Levin JZ, Thompson DA, Amit I, Adiconis X, Fan L, Raychowdhury R, Zeng Q, et al. 2011. Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nat. Biotechnol.* 20:644–652.

- Griffith M, Miller CA, Griffith OL, Krysiak K, Skidmore ZL, Ramu A, Walker JR, Dang HX, Trani L, Larson DE, et al. 2015. Optimizing Cancer Genome Sequencing and Analysis. *Cell Syst.* 1:210–223.
- Gu W, Zhang F, Lupski JR. 2008. Mechanisms for human genomic rearrangements. *Pathogenetics* 1:4.
- Guarino LR. 1978. The Evolution of Abstraction in Programming Languages.
- Gurevich A, Saveliev V, Vyahhi N, Tesler G. 2013. QUASt: Quality assessment tool for genome assemblies. *Bioinformatics* 29:1072–1075.
- Gusev A, Shah MJ, Kenny EE, Ramachandran A, Lowe JK, Salit J, Lee CC, Levandowsky EC, Weaver TN, Doan QC, et al. 2012. Low-pass genome-wide sequencing and variant inference using identity-by-descent in an isolated human population. *Genetics* 190:679–689.
- Haber JE. 2000. Partners and pathways - Repairing a double-strand break. *Trends Genet.* 16:259–264.
- Haffner MC, Aryee MJ, Toubaji A, Esopi DM, Albadine R, Gurel B, Isaacs WB, Bova GS, Liu W, Xu J, et al. 2010. Androgen-induced TOP2B-mediated double-strand breaks and prostate cancer gene rearrangements. *Nat. Genet.* 42:668–675.
- Haque F, Li J, Wu HC, Liang XJ, Guo P. 2013. Solid-state and biological nanopore for real-time sensing of single chemical and sequencing of DNA. *Nano Today* 8:56–74.
- Hayden EC. 2014. Technology: the \$1,000 genome. *Nature* 507:294.
- Healy K. 2007. Nanopore-based single-molecule DNA analysis. *Nanomedicine* 2:459–481.
- van Heesch S, Kloosterman WP, Lansu N, Ruzius FP, Levandowsky E, Lee CC, Zhou S, Goldstein S, Schwartz DC, Harkins TT, et al. 2013. Improving mammalian genome scaffolding using large insert mate-pair next-generation sequencing. *BMC Genomics* 14:257.
- Hernandez D, François P, Farinelli L, Østerås M, Schrenzel J. 2008. De novo bacterial genome sequencing: Millions of very short reads assembled on a desktop computer. *Genome Res.* 18:802–809.
- Hoare CAR. 1962. Quicksort. *Comput. J.* 5:10–16.
- Hogeweg P. 2011. The roots of bioinformatics in theoretical biology. *PLoS Comput. Biol.* 7:e1002021.
- Holt RA, Jones SJM. 2008. The new paradigm of flow cell sequencing. *Genome Res.* 18:839–846.
- Hormozdiari F, Alkan C, Eichler EE, Sahinalp SC. 2009. Combinatorial algorithms for structural variation detection in high-throughput sequenced genomes. *Genome Res.* 19:1270–1278.
- Illumina. 2016. Nextera® Mate Pair Library Prep Reference Guide (15035209 v02). Illumina [Internet]:1–28. Available from: https://support.illumina.com/content/dam/illumina-support/documents/documentation/chemistry_documentation/samplepreps_nextera/nexteramatepair/nextera-mate-pair-reference-guide-15035209-

- 02.pdf%0Ahttp://support.illumina.com/downloads/nextera_xt_sample_prepa
- Illumina. 2019. bcl2fastq2 Conversion Software v2.20. Illumina Softw. Guid.
- Illumina Inc. 2009. Genome Analyzer IIX System. Available from:
https://www.illumina.com/Documents/products/specifications/specification_genome_analyzer.pdf
- Illumina Inc. 2013. Illumina CMOS Chip and One-Channel SBS Chemistry. :1–4.
- ISCN. 2016. ISCN 2016: An International System for Human Cytogenomic Nomenclature (2016). S. Karger Publishing
- Jaccard P. 1901. Étude comparative de la distribution florale dans une portion des Alpes et du Jura. Bull. la Société Vaudoise des Sci. Nat. 37:547–579.
- Jackman SD, Coombe L, Chu J, Warren RL, Vandervalk BP, Yeo S, Xue Z, Mohamadi H, Bohlmann J, Jones SJM, et al. 2018. Tigrint: Correcting assembly errors using linked reads from large molecules. BMC Bioinformatics 19:393.
- Johnson DS, Mortazavi A, Myers RM, Wold B. 2007. Genome-wide mapping of in vivo protein-DNA interactions. Science (80-.). 316:1497–1502.
- Kelley DR, Salzberg SL. 2010. Detection and correction of false segmental duplications caused by genome mis-assembly. Genome Biol. 11:R28.
- Khan AR, Pervez MT, Babar ME, Naveed N, Shoaib M. 2018. A Comprehensive Study of De Novo Genome Assemblers: Current Challenges and Future Prospective. Evol. Bioinforma. 14:1176934318758650.
- Kinsella M, Bafna V. 2012. Combinatorics of the breakage-fusion-bridge mechanism. J. Comput. Biol. 19:662–678.
- Kinsella RJ, Kähäri A, Haider S, Zamora J, Proctor G, Spudich G, Almeida-King J, Staines D, Derwent P, Kerhornou A, et al. 2011. Ensembl BioMarts: A hub for data retrieval across taxonomic space. Database 2011:bar030.
- Kircher M, Witten DM, Jain P, O’roak BJ, Cooper GM, Shendure J. 2014. A general framework for estimating the relative pathogenicity of human genetic variants. Nat. Genet. 46:310–315.
- Köhler S, Carmody L, Vasilevsky N, Jacobsen JOB, Danis D, Gouridine JP, Gargano M, Harris NL, Matentzoglou N, McMurry JA, et al. 2019. Expansion of the Human Phenotype Ontology (HPO) knowledge base and resources. Nucleic Acids Res. 47:D1018–D1027.
- Koltsova AS, Pendina AA, Efimova OA, Chiryaeva OG, Kuznetzova T V., Baranov VS. 2019. On the complexity of mechanisms and consequences of chromothripsis: An update. Front. Genet. 10:393.
- de Koning APJ, Gu W, Castoe TA, Batzer MA, Pollock DD. 2011. Repetitive elements may comprise over Two-Thirds of the human genome. PLoS Genet. 7:e1002384.
- Korbel JO, Campbell PJ. 2013. Criteria for inference of chromothripsis in cancer genomes. Cell 152:1226–1236.
- Korbel JO, Urban AE, Affourtit JP, Godwin B, Grubert F, Simons JF, Kim PM, Palejev D,

- Carriero NJ, Du L, et al. 2007. Paired-end mapping reveals extensive structural variation in the human genome. *Science* (80-.). 318:420–426.
- Köster J, Rahmann S. 2012. Snakemake-a scalable bioinformatics workflow engine. *Bioinformatics* 28:2520–2522.
- Krzywinski M, Schein J, Birol I, Connors J, Gascoyne R, Horsman D, Jones SJ, Marra MA. 2009. Circos: An information aesthetic for comparative genomics. *Genome Res.* 19:1639–1645.
- Kurtzer GM, Sochat V, Bauer MW. 2017. Singularity: Scientific containers for mobility of compute. *PLoS One* 12:e0177459.
- Laduca H, Stuenkel AJ, Dolinsky JS, Keiles S, Tandy S, Pesaran T, Chen E, Gau CL, Palmaer E, Shoaepour K, et al. 2014. Utilization of multigene panels in hereditary cancer predisposition testing: Analysis of more than 2,000 patients. *Genet. Med.*
- Lander ES, Linton LM, Birren B, Nusbaum C, Zody MC, Baldwin J, Devon K, Dewar K, Doyle M, Fitzhugh W, et al. 2001. Initial sequencing and analysis of the human genome. *Nature* 409:860–921.
- Langmead B, Trapnell C, Pop M, Salzberg S. 2009. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol.* [Internet] 10:R25. Available from: papers://1bc19a7c-6e6a-4594-831a-36d8c340e116/Paper/p2438
- Layer RM, Chiang C, Quinlan AR, Hall IM. 2014. LUMPY: A probabilistic framework for structural variant discovery. *Genome Biol.* 15:R84.
- Lee H, Gurtowski J, Yoo S, Nattestad M, Marcus S, Goodwin S, McCombie WR, Schatz M. 2016. Third-generation sequencing and the future of genomics. *bioRxiv*:048603.
- Lee JA, Carvalho CMB, Lupski JR. 2007. A DNA Replication Mechanism for Generating Nonrecurrent Rearrangements Associated with Genomic Disorders. *Cell* 131:1235–1247.
- Lerma MA. 2014. How inefficient can a sort algorithm be? :1–8. Available from: <http://arxiv.org/abs/1406.1077>
- Levene HJ, Korlach J, Turner SW, Foquet M, Craighead HG, Webb WW. 2003. Zero-mode waveguides for single-molecule analysis at high concentrations. *Science* (80-.). 299:682–686.
- Li H. 2012. Exploring single-sample snp and indel calling with whole-genome de novo assembly. *Bioinformatics* 28:1838–1844.
- Li H. 2013. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. *arXiv Prepr. arXiv* [Internet] 00:3. Available from: <http://arxiv.org/abs/1303.3997>
- Li H. 2015. FermiKit: Assembly-based variant calling for Illumina resequencing data. *Bioinformatics* 31:3694–3696.
- Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R. 2009. The Sequence Alignment/Map format and SAMtools. *Bioinformatics* 25:2078–2079.

- Li R, Yu C, Li Y, Lam TW, Yiu SM, Kristiansen K, Wang J. 2009. SOAP2: An improved ultrafast tool for short read alignment. *Bioinformatics* 25:1966–1967.
- Li W, Godzik A. 2006. Cd-hit: A fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics* 22:1658–1659.
- Liu P, Erez A, Nagamani SCS, Dhar SU, Kołodziejska KE, Dharmadhikari A V., Cooper ML, Wiszniewska J, Zhang F, Withers MA, et al. 2011. Chromosome catastrophes involve replication mechanisms generating complex genomic rearrangements. *Cell* 146:889–903.
- Liu S, Huang S, Chen F, Zhao L, Yuan Y, Francis SS, Fang L, Li Z, Lin L, Liu R, et al. 2018. Genomic Analyses from Non-invasive Prenatal Testing Reveal Genetic Associations, Patterns of Viral Infections, and Chinese Population History. *Cell* 175:347–359.
- Lupski JR. 2015. Structural variation mutagenesis of the human genome: Impact on disease and evolution. *Environ. Mol. Mutagen.* 56:419–436.
- Mao Z, Bozzella M, Seluanov A, Gorbunova V. 2008. DNA repair by nonhomologous end joining and homologous recombination during cell cycle in human cells. *Cell Cycle* 7:2902–2906.
- Marçais G, Delcher AL, Phillippy AM, Coston R, Salzberg SL, Zimin A. 2018. MUMmer4: A fast and versatile genome alignment system. *PLoS Comput. Biol.* 14:e1005944.
- Marett L, Jensen JM, Petersen B, Sibbesen JA, Liu S, Villesen P, Skov L, Belling K, Theil Have C, Izarzugaza JMG, et al. 2017. Sequencing and de novo assembly of 150 genomes from Denmark as a population reference. *Nature* 548:87–91.
- Margulies M, Egholm M, Altman WE, Attiya S, Bader JS, Bembien LA, Berka J, Braverman MS, Chen YJ, Chen Z, et al. 2005. Genome sequencing in microfabricated high-density picolitre reactors. *Nature* 437:376–380.
- Mathworks Inc. 2016. MATLAB and statistics toolbox release.
- McClintock B. 1938. The Production of Homozygous Deficient Tissues with Mutant Characteristics by Means of the Aberrant Mitotic Behavior of Ring-Shaped Chromosomes. *Genetics* 23:315–376.
- McClintock B. 1941. The Stability of Broken Ends of Chromosomes in Zea Mays. *Genetics* 26:234–282.
- McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, Garimella K, Altshuler D, Gabriel S, Daly M, et al. 2010. The genome analysis toolkit: A MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res.* 20:1297–1303.
- McLaren W, Gil L, Hunt SE, Riat HS, Ritchie GRS, Thormann A, Flicek P, Cunningham F. 2016. The Ensembl Variant Effect Predictor. *Genome Biol.* 17:122.
- McNaughton AL, Roberts HE, Bonsall D, de Cesare M, Mokaya J, Lumley SF, Golubchik T, Piazza P, Martin JB, de Lara C, et al. 2019. Illumina and Nanopore methods for whole genome sequencing of hepatitis B virus (HBV). *Sci. Rep.* 9:7081.

- Michalet X, Ekong R, Fougerousse F, Rousseaux S, Schurra C, Hornigold N, Van Slegtenhorst M, Wolfe J, Povey S, Beckmann JS, et al. 1997. Dynamic molecular combing: Stretching the whole human genome for high-resolution studies. *Science* (80-). 277:1518–1523.
- Mikkelsen TS, Hillier LW, Eichler EE, Zody MC, Jaffe DB, Yang SP, Enard W, Hellmann I, Lindblad-Toh K, Altheide TK, et al. 2005. Initial sequence of the chimpanzee genome and comparison with the human genome. *Nature* 437:69–87.
- Morris K V., Mattick JS. 2014. The rise of regulatory RNA. *Nat. Rev. Genet.* 15:423–437.
- Mostow J. 1985. Toward Better Models of the Design Process. *AI Mag.* 6:44–44.
- Myers EW, Sutton GG, Delcher AL, Dew IM, Fasulo DP, Flanigan MJ, Kravitz SA, Mobarry CM, Reinert KHJ, Remington KA, et al. 2000. A whole-genome assembly of *Drosophila*. *Science* (80-). 287:2196–2204.
- Nagalakshmi U, Wang Z, Waern K, Shou C, Raha D, Gerstein M, Snyder M. 2008. The transcriptional landscape of the yeast genome defined by RNA sequencing. *Science* (80-). 320:1344–1349.
- Narzisi G, ORawe J a., Iossifov I, Fang H, Lee Y -h., Wang Z, Wu Y, Lyon GJ, Wigler M, Schatz MC. 2013. Scalpel: Accurate detection of de novo and transmitted INDELs within exome-capture data using micro-assembly. *bioRxiv* [Internet]. Available from: <http://biorxiv.org/content/early/2014/04/15/001370.abstract>
- Nattestad M, Schatz MC. 2016. Assemblytics: A web analytics tool for the detection of variants from an assembly. *Bioinformatics* 32:3021–3023.
- Nederbragt AJ, Rounge TB, Kausrud KL, Jakobsen KS. 2010. Identification and Quantification of Genomic Repeats and Sample Contamination in Assemblies of 454 Pyrosequencing Reads. *Sequencing*.
- Van Nieuwerburgh F, Thompson RC, Ledesma J, Deforce D, Gaasterland T, Ordoukhanian P, Head SR. 2012. Illumina mate-paired DNA sequencing-library preparation using Cre-Lox recombination. *Nucleic Acids Res.* 43:e24–e24.
- Oliphant T, Millma J k. 2006. A guide to NumPy. *Trelgol Publ.*
- Onmus-Leone F, Hang J, Clifford RJ, Yang Y, Riley MC, Kuschner RA, Waterman PE, Lesho EP. 2013. Enhanced De Novo Assembly of High Throughput Pyrosequencing Data Using Whole Genome Mapping. *PLoS One* 8:e61762.
- Oualline S, Nye EA, Dougherty D. 1997. *Practical C ++ Programming*.
- Pellestor F. 2019. Chromoanagenesis: Cataclysms behind complex chromosomal rearrangements. *Mol. Cytogenet.* 12:6.
- Peltola H, Söderlund H, Ukkonen E. 1984. SEQAID: A DNA sequence assembling program based on a mathematical model. *Nucleic Acids Res.* 12:307–321.
- Plauger PJ. 2002. *The C/C++ programming language*.
- Prechelt L. 2003. Are Scripting Languages Any Good? A Validation of Perl, Python, REXX, and Tcl against C, C++, and Java. *Adv. Comput.* 57:205–270.
- Prüfer K, Munch K, Hellmann I, Akagi K, Miller JR, Walenz B, Koren S, Sutton G, Kodira

- C, Winer R, et al. 2012. The bonobo genome compared with the chimpanzee and human genomes. *Nature* 486:527–531.
- Purushothaman S, Toumazou C, Ou CP. 2006. Protons and single nucleotide polymorphism detection: A simple use for the Ion Sensitive Field Effect Transistor. *Sensors Actuators, B Chem.* 114:964–968.
- Quail MA, Smith M, Coupland P, Otto TD, Harris SR, Connor TR, Bertoni A, Swerdlow HP, Gu Y. 2012. A tale of three next generation sequencing platforms: comparison of Ion Torrent, Pacific Biosciences and Illumina MiSeq sequencers. *BMC Genomics* 13:341.
- Quinlan AR, Hall IM. 2010. BEDTools: A flexible suite of utilities for comparing genomic features. *Bioinformatics* 26:841–842.
- Ramey C, University CWR, Fox B, Foundation FS. 2009. GNU bash. Ref. Doc. Bash.
- Rang FJ, Kloosterman WP, de Ridder J. 2018. From squiggle to basepair: Computational approaches for improving nanopore sequencing read accuracy. *Genome Biol.* 90:90.
- Rausch T, Zichner T, Schlattl A, Stütz AM, Benes V, Korbel JO. 2012. DELLY: Structural variant discovery by integrated paired-end and split-read analysis. *Bioinformatics* 28:i333–i339.
- Redin D, Borgström E, He M, Aghelpasand H, Käller M, Ahmadian A. 2017. Droplet Barcode Sequencing for targeted linked-read haplotyping of single DNA molecules. *Nucleic Acids Res.* 45:e125–e125.
- Redon R, Ishikawa S, Fitch KR, Feuk L, Perry GH, Andrews TD, Fiegler H, Shapero MH, Carson AR, Chen W, et al. 2006. Global variation in copy number in the human genome. *Nature* 444:444–454.
- Rhodes J, Beale MA, Fisher MC. 2014. Illuminating choices for library prep: A comparison of library preparation methods for whole genome sequencing of *Cryptococcus neoformans* using Illumina HiSeq. *PLoS One* 9:e113501.
- Rossum G Van, Drake FL. 1995. Python Tutorial, Technical Report CS-R9526. In: Centrum voor Wiskunde en Informatica (CWI).
- Sahlén P, Abdullayev I, Ramsköld D, Matskova L, Rilakovic N, Lötstedt B, Albert TJ, Lundeberg J, Sandberg R. 2015. Genome-wide mapping of promoter-anchored interactions with close to single-enhancer resolution. *Genome Biol.* 16:156.
- Sahlin K, Vezzi F, Nystedt B, Lundeberg J, Arvestad L. 2014. BESST - Efficient scaffolding of large fragmented assemblies. *BMC Bioinformatics* 15:281.
- Salzberg SL. 2018. Open questions: How many genes do we have? *BMC Biol.* 16:94.
- Sanner MF. 1999. Python: A programming language for software integration and development. *J. Mol. Graph. Model.* 17:57–61.
- Schalamun M, Nagar R, Kainer D, Beavan E, Eccles D, Rathjen JP, Lanfear R, Schwessinger B. 2019. Harnessing the MinION: An example of how to establish long-read sequencing in a laboratory using challenging plant tissue from *Eucalyptus pauciflora*. *Mol. Ecol. Resour.* 19:77–89.
- Schneider VA, Graves-Lindsay T, Howe K, Bouk N, Chen HC, Kitts PA, Murphy TD,

- Pruitt KD, Thibaud-Nissen F, Albracht D, et al. 2017. Evaluation of GRCh38 and de novo haploid genome assemblies demonstrates the enduring quality of the reference assembly. *Genome Res.* 17:849–864.
- Schwartz D, Li X, Hernandez L, Ramnarain S, Huff E, Wang Y. 1993. Ordered restriction maps of *Saccharomyces cerevisiae* chromosomes constructed by optical mapping. *Science* (80-.). 262:110–114.
- Scriven PN, Handyside AH, Mackie Ogilvie C. 1998. Chromosome translocations: Segregation modes and strategies for preimplantation genetic diagnosis. *Prenat. Diagn.* 18:1437–1449.
- Sedlazeck FJ, Rescheneder P, Smolka M, Fang H, Nattestad M, Von Haeseler A, Schatz MC. 2018. Accurate detection of complex structural variations using single-molecule sequencing. *Nat. Methods* 18:461–468.
- Seo JS, Rhie A, Kim Junsoo, Lee S, Sohn MH, Kim CU, Hastie A, Cao H, Yun JY, Kim Jihye, et al. 2016. De novo assembly and phasing of a Korean human genome. *Nature* 538:243–247.
- Shang J, Zhu F, Vongsangnak W, Tang Y, Zhang W, Shen B. 2014. Evaluation and comparison of multiple aligners for next-generation sequencing data analysis. *Biomed Res. Int.* 2014.
- Sharp AJ, Locke DP, McGrath SD, Cheng Z, Bailey JA, Vallente RU, Pertz LM, Clark RA, Schwartz S, Segraves R, et al. 2005. Segmental duplications and copy-number variation in the human genome. *Am. J. Hum. Genet.* 77:78–88.
- Shaw CJ. 2004. Implications of human genome architecture for rearrangement-based disorders: the genomic basis of disease. *Hum. Mol. Genet.* 13:R57–R64.
- Shearer LA, Anderson LK, de Jong H, Smit S, Goicoechea JL, Roe BA, Hua A, Giovannoni JJ, Stack SM. 2014. Fluorescence in situ hybridization and optical mapping to correct scaffold arrangement in the tomato genome. *G3 Genes, Genomes, Genet.* 4:1395–1405.
- Shendure J, Ji H. 2008. Next-generation DNA sequencing. *Nat. Biotechnol.* 20:1297–1303.
- Sherman RM, Forman J, Antonescu V, Puiu D, Daya M, Rafaels N, Boorgula MP, Chavan S, Vergara C, Ortega VE, et al. 2019. Assembly of a pan-genome from deep sequencing of 910 humans of African descent. *Nat. Genet.* 51:30–35.
- Simpson JT, Durbin R. 2012. Efficient de novo assembly of large genomes using compressed data structures. *Genome Res.* 22:549–556.
- Simpson JT, Wong K, Jackman SD, Schein JE, Jones SJM, Birol I. 2009. ABySS: A parallel assembler for short read sequence data. *Genome Res.* 19:1117–1123.
- Sinkov A, Saxon JA, Plette WS. 1963. Programming the IBM 1401: A Self-Instructional Programmed Manual. *Math. Comput.*
- Soare RI. 2009. Turing oracle machines, online computing, and three displacements in computability theory. *Ann. Pure Appl. Log.* 160:368–399.
- Stoddart D, Heron AJ, Mikhailova E, Maglia G, Bayley H. 2009. Single-nucleotide discrimination in immobilized DNA oligonucleotides with a biological nanopore.

- Proc. Natl. Acad. Sci. U. S. A. 106:7702–7707.
- Stoeckius M, Hafemeister C, Stephenson W, Houck-Loomis B, Chattopadhyay PK, Swerdlow H, Satija R, Smibert P. 2017. Simultaneous epitope and transcriptome measurement in single cells. *Nat. Methods* 14:865–868.
- Stranneheim H, Engvall M, Naess K, Lesko N, Larsson P, Dahlberg M, Andeer R, Wredenberg A, Freyer C, Barbaro M, et al. 2014. Rapid pulsed whole genome sequencing for comprehensive acute diagnostics of inborn errors of metabolism. *BMC Genomics* [Internet] 15:1090. Available from: <http://bmcgenomics.biomedcentral.com/articles/10.1186/1471-2164-15-1090>
- Stroustrup B. 1985. C plus plus TUTORIAL. In: *Proceedings of the Annual Conference of the Association for Computing Machinery*.
- Sudmant PH, Rausch T, Gardner EJ, Handsaker RE, Abyzov A, Huddleston J, Zhang Y, Ye K, Jun G, Fritz MHY, et al. 2015. An integrated map of structural variation in 2,504 human genomes. *Nature* 56:75–81.
- Tan G, Opitz L, Schlapbach R, Rehrauer H. 2019. Long fragments achieve lower base quality in Illumina paired-end sequencing. *Sci. Rep.* 9:2856.
- Tarasov A, Vilella AJ, Cuppen E, Nijman IJ, Prins P. 2015. Sambamba: Fast processing of NGS alignment formats. *Bioinformatics* 31:2032–2034.
- Taschner PEM, den Dunnen JT. 2011. Describing structural changes by extending HGVS sequence variation nomenclature. *Hum. Mutat.* 32:507–511.
- Tattini L, D’Aurizio R, Magi A. 2015. Detection of Genomic Structural Variants from Next-Generation Sequencing Data. *Front. Bioeng. Biotechnol.* [Internet] 3. Available from: <http://journal.frontiersin.org/Article/10.3389/fbioe.2015.00092/abstract>
- Thankaswamy-Kosala S, Sen P, Nookaew I. 2017. Evaluation and assessment of read-mapping by multiple next-generation sequencing aligners based on genome-wide characteristics. *Genomics* 109:186–191.
- DI Tommaso P, Chatzou M, Floden EW, Barja PP, Palumbo E, Notredame C. 2017. Nextflow enables reproducible computational workflows. *Nat. Biotechnol.* 35:316–319.
- Tümer Z, Tommerup N, Tønnesen T, Kreuder J, Craig IW, Horn N. 1992. Mapping of the Menkes locus to Xq13.3 distal to the X-inactivation center by an intrachromosomal insertion of the segment Xq13.3-q21.2. *Hum. Genet.* 88:668–672.
- Venter JC, Adams MD, Sutton GG, Kerlavage AR, Smith HO, Hunkapiller M. 1998. Shotgun sequencing of the human genome. *Science* (80-.). 280:1540–1542.
- Vogt V. 1997. *Retroviral Virions and Genomes*.
- Warren RL, Yang C, Vandervalk BP, Behsaz B, Lagman A, Jones SJM, Birol I. 2015. LINKS: Scalable, alignment-free scaffolding of draft genomes with long reads. *Gigascience* 4:35.
- Weckselblatt B, Rudd MK. 2015. Human Structural Variation: Mechanisms of Chromosome Rearrangements. *Trends Genet.* 31:587–599.
- Weisenfeld NI, Kumar V, Shah P, Church DM, Jaffe DB. 2017. Direct determination of

- diploid genome sequences. *Genome Res.* 27:757–767.
- Ye K, Schulz MH, Long Q, Apweiler R, Ning Z. 2009. Pindel: A pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics* 25:2865–2871.
- Zakov S, Bafna V. 2014. Reconstructing breakage fusion bridge architectures using noisy copy numbers. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*.
- Zeigler Allen L, McCrow JP, Ininbergs K, Dupont CL, Badger JH, Hoffman JM, Ekman M, Allen AE, Bergman B, Venter JC. 2017. The Baltic Sea Virome: Diversity and Transcriptional Activity of DNA and RNA Viruses. *mSystems* 2:e00125-16.
- Zepeda-Mendoza CJ, Morton CC. 2019. The Iceberg under Water: Unexplored Complexity of Chromoanagenesis in Congenital Disorders. *Am. J. Hum. Genet.* 104:565–577.
- Zerbino DR, Birney E. 2008. Velvet: Algorithms for de novo short read assembly using de Bruijn graphs. *Genome Res.* 18:821–829.
- Zhang CZ, Leibowitz ML, Pellman D. 2013. Chromothripsis and beyond: Rapid genome evolution from complex chromosomal rearrangements. *Genes Dev.* 27:2513–2530.
- Zhang CZ, Spektor A, Cornils H, Francis JM, Jackson EK, Liu S, Meyerson M, Pellman D. 2015. Chromothripsis from DNA damage in micronuclei. *Nature* 522:179–184.
- Zhang F, Christiansen L, Thomas J, Pokholok D, Jackson R, Morrell N, Zhao Y, Wiley M, Welch E, Jaeger E, et al. 2017. Haplotype phasing of whole human genomes using bead-based barcode partitioning in a single tube. *Nat. Biotechnol.* 35:852–857.
- Zhang F, Khajavi M, Connolly AM, Towne CF, Batish SD, Lupski JR. 2009. The DNA replication FoSTeS/MMBIR mechanism can generate genomic, genic and exonic complex rearrangements in humans. *Nat. Genet.* 41:849–853.
- Zheng D, Frankish A, Baertsch R, Kapranov P, Reymond A, Siew WC, Lu Y, Denoeud F, Antonarakis SE, Snyder M, et al. 2007. Pseudogenes in the ENCODE regions: Consensus annotation, analysis of transcription, and evolution. *Genome Res.* 17:839–851.
- Zheng GXY, Lau BT, Schnall-Levin M, Jarosz M, Bell JM, Hindson CM, Kyriazopoulou-Panagiotopoulou S, Masquelier DA, Merrill L, Terry JM, et al. 2016. Haplotyping germline and cancer genomes with high-throughput linked-read sequencing. *Nat. Biotechnol.* 34:303–311.
- Zimin A V., Delcher AL, Florea L, Kelley DR, Schatz MC, Puiu D, Hanrahan F, Pertea G, Van Tassell CP, Sonstegard TS, et al. 2009. A whole-genome assembly of the domestic cow, *Bos taurus*. *Genome Biol.* 10:R42.
- Zook JM, Chapman B, Wang J, Mittelman D, Hofmann O, Hide W, Salit M. 2014. Integrating human sequence data sets provides a resource of benchmark SNP and indel genotype calls. *Nat. Biotechnol.* 32:246–251.